TM-71-1022-4

TECHNICAL MEMORANDUM

DESIGN OF CONTROL LAWS FOR CONTROL MOMENT GYROSCOPES WITH APPLICATION TO SKYLAB

Bellcomm

BELLCOMM, INC.
955 L'ENFANT PLAZA NORTH, S.W. WASHINGTON, D.C. 20024

COVER SHEET FOR TECHNICAL MEMORANDUM

TITLE- Design of Control Laws for Control Moment Gyroscopes with Application to Skylab

TM- 71-1022-4

DATE- September 3, 1971

FILING CASE NO(S)- 620

AUTHOR(S)- S. C. Chu
J. Kranton

FILING SUBJECT(S)-
(ASSIGNED BY AUTHOR(S)- Attitude Control,
Control Moment Gyroscopes

ABSTRACT

A rigorous treatment of CMG gimbal angle control is presented in the framework of an optimization problem. Two algorithms for control are developed and implementation questions are discussed.

The CMG control law being developed by MSFC for the Skylab is explained in terms of the basic components of an optimal solution. Further, the gimbal angle trajectories of the MSFC control law appears to be reproducible by a proper choice of the penalty function in the optimization problem.

Savings in computer memory storage can be achieved by employing one of the control laws developed here. IBM, Huntsville, has shown that a saving of 224 words could be realized. However, with the ATM digital computer, execution time would be increased. Should the need for additional memory become compelling, consideration can be given to replacing the MSFC law by one presented here.

Bellcomm

955 L'Enfant Plaza North, S.W.
Washington, D. C. 20024

date: September 3, 1971

to: Distribution

TM-71-1022-4

from: S. C. Chu, J. Kranton

subject: Design of Control Laws for Control Moment Gyroscopes with Application to Skylab - Case 620

TECHNICAL MEMORANDUM

Introduction

The design of control laws for active control-moment-gyro (CMG) systems divides naturally into two parts - (1) specifying the desired control torque T^d as a function of the attitude and rate error of the spacecraft and (2) controlling the rate of change of CMG spin angular momentum $\dot{H}$ such that the actual control torque T^a equals the desired T^d. This memorandum deals with the second problem. The first has been investigated in Ref. 1, which contains much historical background.

For any physical arrangement of CMGs, H can be written as*

$$\dot{H} = C\left(\delta^a\right)\dot{\delta}^a = T^a - \omega \times H\left(\delta^a\right)$$

where, for a system of N two degrees-of-freedom CMGs, δ^a is a $2N \times 1$ vector of actual gimbal angles, $C(\delta^a) = \{\partial H_i/\partial \delta^a{}_j\}$** is a $3 \times 2N$ matrix whose elements are trigonometric functions of the gimbal angles, and ω is the spacecraft angular velocity. The magnitude of the spin angular momentum of each CMG is assumed to be the same and each term in this equation is a torque per unit of spin angular momentum of one CMG.

---

*The effect of non-spin angular momentum terms are negligible [Ref. 1], and therefore are not included.

**H_i (i=1,2,3) and δ^a_j (j=1,...,2N) are the components of the vectors H and δ^a, respectively.

A servomechanism on each gimbal axis controls the actual gimbal angle rates $\dot{\delta}^a$ by tracking commanded gimbal angle rates $\dot{\delta}^c$ generated in the control computer. The response time of these servomechanisms is sufficiently small so that $\dot{\delta}^a$ tracks $\dot{\delta}^c$ very closely, close enough in fact that the first step is to replace $\dot{\delta}^a$ by $\dot{\delta}^c$ in the equation for $\dot{H}$.

$$\dot{H} = C\left(\delta^a\right)\dot{\delta}^c = T^a - \omega \times H\left(\delta^a\right)$$

In what follows the superscripts a and c on δ will be suppressed, but it is to be understood that in the control computer $\dot{\delta}^c$ is calculated with the matrix C and the vector H determined from actual gimbal angles (or sines and cosines of the actual gimbal angles).

Implementation of CMG control is likely, as on the Skylab, to employ a digital computer. A generic block diagram for such a system is shown on Figure 1. At time $t = \tau_k$, sample values of attitude and rate error are used to calculate a sample value of the desired CMG torque T^d from which a sample value of $\dot{\delta}$ is calculated. This $\dot{\delta}$ is held constant until the next sampling time and is applied as input to the servomechanisms that control gimbal angle rates.

The purpose of this report is to develop, in a unified and systematic way, the underlying principles for determining the commanded rates $\dot{\delta}$, in the framework of an optimization problem. Computational algorithms for obtaining such rates are derived and the computational problems involved in implementation are discussed. The Marshall Space Flight Center (MSFC) control law being implemented on the Skylab is also explained in terms of the principles developed here.

I. Mathematical Statement of the Problem

For the sake of definiteness, we shall confine ourselves in this paper to the case of 3 CMGs (N=3). In this event, the gimbal angle vector δ is a 6-vector and is given by

$$\delta' = \left(\beta_1, \alpha_1, \beta_2, \alpha_2, \beta_3, \alpha_3\right)$$

where the β_i and α_i represent the inner and outer gimbal angles, respectively.

The momentum vector, H, is given by $H = h_1 + h_2 + h_3$, where h_i is the unit momentum of each gyro, and in the case of Skylab (Ref. 2):

$$h_1 = \begin{pmatrix} \cos\alpha_1\cos\beta_1 \\ -\sin\alpha_1\cos\beta_1 \\ -\sin\beta_1 \end{pmatrix}, \quad h_2 = \begin{pmatrix} -\sin\beta_2 \\ \cos\alpha_2\cos\beta_2 \\ -\sin\alpha_2\cos\beta_2 \end{pmatrix}, \quad h_3 = \begin{pmatrix} -\sin\alpha_3\cos\beta_3 \\ -\sin\beta_3 \\ \cos\alpha_3\cos\beta_3 \end{pmatrix}$$

Hence, the 3×6 matrix $C(\delta)$ is given by

$$C = \left(\begin{array}{c|c|c|c|c|c} -\cos\alpha_1\sin\beta_1 & -\sin\alpha_1\cos\beta_1 & -\cos\beta_2 & 0 & \sin\alpha_3\sin\beta_3 & -\cos\alpha_3\cos\beta_3 \\ \sin\alpha_1\sin\beta_1 & -\cos\alpha_1\cos\beta_1 & -\cos\alpha_2\sin\beta_2 & -\sin\alpha_2\cos\beta_2 & -\cos\beta_3 & 0 \\ -\cos\beta_1 & 0 & \sin\alpha_2\sin\beta_2 & -\cos\alpha_2\cos\beta_2 & -\cos\alpha_3\sin\beta_3 & -\sin\alpha_3\cos\beta_3 \end{array}\right)$$

We can now formulate the problem mathematically: Given $\delta(t)$, the CMGs deliver a torque, T^a, according to the relation:

$$C\left(\delta(t)\right)\dot{\delta}(t) + \omega(t) \times H\left(\delta(t)\right) = T^a \qquad (1)$$

We require the controller to act in such a way so that both $\dot{\delta}$ and T^d are constant in each sampling interval $[t_k, t_{k+1}]$; i.e., we seek a $\delta(t)$ which satisfies:

$$C\left(\delta(t)\right)\dot{\delta}\left(t_k\right) + \omega \times H\left(\delta(t)\right) = T_k^d \quad , \quad \text{for } t \text{ in } [t_k, t_{k+1}] \qquad (2)$$

where $\{T_k^d\}$ is a sequence of constant vectors. Furthermore, we wish to minimize some penalty function $J(\delta(t))$ subject to the constraint Eq. (2).

The role of the penalty function is two-fold. First, because of the fact that Eq. (2) is a system of 3 equations in 6 unknowns, minimizing the functional J enables us to determine a specific (although not necessarily unique) solution δ. Second,

the minimization also enables us to incorporate certain desirable properties into the solution δ, such as smoothness of trajectory, avoidance of gimbal stops, etc. These properties will be discussed further below.

The constraint relation given by Eq. (2) cannot, in general, be satisfied. For, if $\dot{\delta}$ is constant in the interval $[t_k, t_{k+1}]$, then $\delta(t) = \delta(t_k) + \dot{\delta}(t_k)(t-t_k)$, and the left-hand-side of Eq. (2) will be varying with time while the right-hand-side is constant.

In Section II, we give two approaches for solving the optimization problem approximately. In each, we shall replace the constraint condition (Eq. (2)) by another equation which is realizable. In the first approach, we shall demand that the constraint be satisfied only at the sampling points. In the second, we require that an integrated version of Eq. (2) be satisfied at the sampling points. For both, we shall give a description along with an analytical expression for their errors. Algorithms for the computation of angles and rates are derived for the general class of quadratic penalty functions. Some stability questions are also addressed. Specific case studies are then presented, (including duplication of the gimbal angle trajectories obtained at MSFC) showing some of the salient features of each of these approaches. Their relative merits are also compared.

II. Two Approaches to Solving the Minimization Problem

A. Torque Matching at the Sampling Times

At the sampling time $t = t_k$, we have the current gimbal angles δ_k, spacecraft rates ω_k, the momentum vector $H(\delta_k) \equiv H_k$, the matrix $C(\delta_k) \equiv C_k$ and the desired torque T_k^d. We seek a set of gimbal angles δ_{k+1} which minimizes a cost function $J(\delta_{k+1}, \delta_k)$, (which may depend on δ_k, as well as on δ_{k+1}), subject to the constraint

$$\frac{1}{\Delta t} C_k\left(\delta_{k+1} - \delta_k\right) + \omega_k \times H_k = T_k^d \quad , \tag{3}$$

where $\Delta t = t_{k+1} - t_k$. We assume for $k = 0$, δ_0 is defined and satisfies suitable predetermined properties.

In this fashion, a sequence $\{\delta_k\}$ is determined. Next, define the sequence $\{\dot{\delta}_k\}$ by

$$\dot{\delta}_k \equiv \frac{\delta_{k+1} - \delta_k}{\Delta t} \tag{4}$$

Finally we define the vector function $\delta(t)$ for t in the interval $[t_k, t_{k+1}]$ by

$$\delta(t) \equiv \delta_k + \dot{\delta}_k\left(t-t_k\right) \tag{5}$$

Should the gimbal angles, δ, evolve according to this trajectory, the actual torque, T^a, produced in the interval $[t_k, t_{k+1}]$ would be, according to Eq. (1):

$$T^a = C\left(\delta(t)\right)\dot{\delta}_k + \omega(t) \times H(t)$$

Thus,

$$T^a = T^d = T^d_k \quad \text{when } t = t_k$$

but, in general,

$$T^a \neq T^d \quad \text{when } t_k < t < t_{k+1}$$

One can obtain an expression for the mean, or average, error of T^a over the interval $[t_k, t_{k+1}]$. The mean error is

$$\frac{1}{\Delta t}\int_{t_k}^{t_{k+1}}\left(T^a - T^d_k\right)d\tau = \frac{1}{\Delta t}\int_{t_k}^{t_{k+1}}\left[\left(C\left(\delta(\tau)\right) - C_k\right)\dot{\delta}_k + \left(\omega(\tau)\times H\left(\delta(\tau)\right) - \omega_k \times H_k\right)\right]d\tau \tag{6}$$

This is a vector equation. If we examine a typical component, say the $i^{\underline{th}}$, and apply the mean value theorem of integral calculus, we obtain

$$\frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \left(T^a - T_k^d\right)^i d\tau = \left(C\left(\delta_k + \dot{\delta}_k \varepsilon_k\right) - C_k\right)\dot{\delta}_k + \left(\omega\left(t_k+\varepsilon_k\right) \times H\left(\delta_k + \dot{\delta}_k \varepsilon_k\right) - \omega_k \times H_k\right) \tag{7}$$

for some ε_k such that $0 < \varepsilon_k < \Delta t$.

In the case where the penalty function J is a quadratic in δ_{k+1}, the obtained algorithms for δ_{k+1} are easily implemented on the digital computer. For the sake of brevity, let us first introduce the notation

$$\bar{\delta}_k = \delta_k - \hat{\delta}$$

for some constant reference vector $\hat{\delta}$. (For example, in the case of the Skylab CMGs, $\hat{\delta} = (0,\pi/4,0,\pi/4,0,\pi/4)$.)

Suppose now, J has the form

$$J\left(\delta_{k+1}, \delta_k\right) = \frac{1}{2}\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right)' \; Q\left(\delta_k\right)\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right) \tag{8}$$

where s is a parameter, $0 \leq s \leq 1$, and Q is a positive definite matrix, which may depend on the current angles δ_k. The motivation for writing a quadratic J in this form is that

$$\bar{\delta}_{k+1} - s\bar{\delta}_k = (1-s)\bar{\delta}_{k+1} + s\left(\bar{\delta}_{k+1} - \bar{\delta}_k\right) = (1-s)\bar{\delta}_{k+1} + s\left(\delta_{k+1} - \delta_k\right)$$

which is a linear combination of angles ($\bar{\delta}_{k+1}$) and rates ($\delta_{k+1} - \delta_k$), and s is a measure of the rate mixture. Q can be thought of as a weighting matrix. (An example of where unequal weights may be used is seen in the case of the Skylab. There the angles β_i traverse between approximately $\pm\pi/2$ while the α_i traverse between $\pm\pi$. In order to equalize this difference in dynamic ranges, we can take Q to be a diagonal matrix with the diagonal elements to be (1, .25, 1, .25, 1, .25).) Minimizing J in this form is equivalent to minimizing the general quadratic form

$$J = \frac{1}{2}\,\bar{\delta}_{k+1}'\,Q_1(\delta_k)\bar{\delta}_{k+1} + \bar{\delta}_{k+1}'\,Q_2(\delta_k)\nu$$

for suitable non-singular matrices Q_1 and Q_2 and vector ν.

The problem, for the penalty function given in Eq. (8), can be stated as: given the current angles δ_k, rates ω_k, momentum H_k, the weighting matrix $Q_k \equiv Q(\delta_k)$, the parameter s, and the desired torque T_k^d, find $\bar{\delta}_{k+1}$ which minimizes (Eq. (8))

$$J = \frac{1}{2}\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right)' Q_k \left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right)$$

subject to the constraint

$$\frac{1}{\Delta t} C_k\left(\delta_{k+1} - \delta_k\right) = T_k^d - \omega_k \times H_k \equiv T_k^c \tag{9}$$

An application of the Lagrange multiplier method yields the formula

$$\begin{aligned}\bar{\delta}_{k+1} - \bar{\delta}_k = (1-s)\left[Q_k^{-1}C_k'\left(C_kQ_k^{-1}C_k'\right)^{-1}C_k - I\right]\bar{\delta}_k \\ + Q_k^{-1}C_k'\left(C_kQ_k^{-1}C_k'\right)^{-1}T_k^c\Delta t\end{aligned} \tag{10}$$

This formula is valid if and only if the matrix $(C_kQ^{-1}C_k')$ is non-singular for each k. If the Q_k are positive definite, then the $C_kQ_k^{-1}C_k'$ are non-singular if and only if the momentum vectors of the CMGs are not colinear. This assertion is proved in Appendix 1.

Equation (10) yields, at once, the gimbal angle rates to be commanded for the time interval $[t_k, t_{k+1}]$, and, therefore, the optimal angles δ_{k+1} to be attained at $t = t_{k+1}$.

A natural "stability" question for this kind of system is: What happens to the gimbal angles when the desired torque becomes zero at a certain time t_N and remains zero thereafter? This is discussed in Appendix 3.

We now discuss some typical case studies using this approach. Computer runs were made with a program based on the algorithm given by Eq. (10), with $\hat{\delta}' = (0,\pi/4,0,\pi/4,0,\pi/4)$. The case studies included both "open-loop" and "closed-loop" models. In the first case, the driving torque T^d is assumed to be just the gravity-gradient torque. In the closed-loop case, the torque is obtained, through feedback, by computing the attitude and rate errors.

1. Open-loop Runs

In these runs, we have assumed $\omega = 0$ for all t. Thus, the average error of the torque over the interval $[t_k, t_{k+1}]$, by Eq. (6), is

$$\frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \left(C\left(\delta(\tau)\right) - C_k\right)\dot{\delta}_k d\tau$$

and the total error up to time t_{k+1} is

$$\int_0^{t_{k+1}} C\left(\delta(\tau)\right)\dot{\delta}_k d\tau - \sum_{j=0}^{k+1} C_j\dot{\delta}_j\Delta t = H_{k+1} - \sum_{j=0}^{k+1} C_j\dot{\delta}_j\Delta t \qquad (11)$$

We also take the Skylab solar inertial attitude, with the solar elevation angle, β, equal to 30°, and starting at 45° before orbital noon. The matrix Q is taken to be the identity matrix, and $\bar{\delta}_0$ is taken to be the zero vector.

Figure 2 compares the gimbal angles obtained for s = 0 (i.e., no rate term in J) and Δt = .2 sec. and Δt = 1 sec. Figure 3 represents the total errors in torque for these two cases. Notice that as Δt becomes smaller, the oscillations become smaller in amplitude and greater in frequency. Of greater importance, note that the total error in torque begins to be significant in the vicinity of 400 seconds into the orbit and grows larger as time goes on. This error seems independent of the size of the sampling intervals. It turns out that the matrices involved (in Eq. (10)) become very sensitive to small

changes in δ near t = 400 sec., the time near which the oscillations first appear. These phenomena occurred in other cases and for other values of Δt which are not included in this report.

If we now start adding the rate term in the penalty function by increasing s, we find that the oscillations start to disappear, and the total errors in torque decrease dramatically. Figures 4 and 5 compare runs for s = .5 and .75 with Δt = .2 sec. We note also that there is a range of values of s (between .5 and .9) where the gimbal angles obtained are not greatly changed with changes in s. These results agree quite well with those obtained by the second approach to be described below.

2. Closed-loop Runs

In these runs the desired torque is a linear function of the attitude and rate errors of the spacecraft, as it will be on the Skylab.

In order to cause the momentum vector to swing over its maximum range, the gravity gradient torque on the Skylab was artificially scaled up. The scale factors used were

β	Torque Factor
0	2.06
±30°	1.73
±45°	1.63
±60°	1.66

where β is the solar elevation angle from the orbital plane.

Runs were made using three control laws: the torque matching control law given by Eq. (10), the MSFC control law, and another version of the torque matching law. This last version produced gimbal angle trajectories nearly identical with those of the MSFC control law.

In the closed-loop case, the torque matching law displayed sharply varying, jagged gimbal angle trajectories for Δt = .2 sec. (the Skylab value) and s < .8. A set of trajectories for s = .9 and s = .2 is shown in Figure 6. This illustrates, once more, that increasing the rate term mixture in the penalty function smoothes out the gimbal angle trajectories.

The MSFC control law is conceptually of the torque matching type. It also produces jagged gimbal angle trajectories unless a gain factor associated with it is set low. This factor plays a similar role to (1-s) in Eq. (10), as we will now show.

Notice that the first term on the right side of Eq. (10) produces zero torque (i.e., it is a vector in the null space of C_k). The second term produces the torque. This is just an example of the fact that the solution to a consistent set of linear equations (i.e., Eq. (9)) consists of a vector from the null space plus a particular solution.

Now, the MSFC control law (Ref. 2) is the sum of two parts, the "rotation law" and the "steering law". The rotation law generates gimbal angle rates, say $\dot{\delta}^R$, that produce no torque. Hence, $\dot{\delta}^R$ is in the null space of C_k. The steering law produces gimbal angle rates, say $\dot{\delta}^S$, that provide the desired torque. Thus, $\dot{\delta}^S$ is a particular solution of Eq. (9). Associated with the rotation law is a scale factor called K_R (Ref. 2); that is, the total gimbal angle rate is $K_R\dot{\delta}^R + \dot{\delta}^S$. We see that the factor K_R plays the same role as (1-s) in Eq. (10) by scaling down the null space component of the commanded gimbal angle rates. The value of K_R given in Ref. 2 is 0.25, which was determined empirically in simulations at MSFC. The tendency for jagged gimbal angle trajectories was found to increase with increasing K_R, just as this tendency increased with increasing (1-s) using the torque matching algorithm given by Eq. (10). This fact is displayed in Figure 7 by the discontinuities in the gimbal angle trajectories.

Since the MSFC law has a similar structure to Eq. (10) (i.e., the sum of a null solution and a particular solution) a question that comes to mind is, "Can the MSFC law be derived as a solution to an optimization problem?" Our attempt to do so proved too formidable to pursue at this time. The next question then is: Can a penalty function, (Eq. (8)), be chosen such that the resulting algorithm produces gimbal angle trajectories identical, or very similar, to those of the MSFC law? The answer to this question is yes.

If the penalty function used is

$$J = \frac{1}{2}\left(\bar{\delta}_{k+1}-\bar{\delta}_k\right)' Q_1 \left(\bar{\delta}_{k+1}-\bar{\delta}_k\right) + (1-s)\left(\bar{\delta}_{k+1}-\bar{\delta}_k\right)' Q_2 \nu$$

the solution to the optimization problem is

$$\bar{\delta}_{k+1} - \bar{\delta}_k = (1-s)\left[Q_1^{-1}C_k'\left(C_kQ_1^{-1}C_k'\right)^{-1}C_k - I\right]Q_1^{-1}Q_2\nu + Q_1^{-1}C_k'\left(C_kQ_1^{-1}C_k'\right)^{-1}T_k^c\Delta t \tag{10a}$$

The elements of Q_1, Q_2, and ν are chosen using the following rationale. Let Q_1 and Q_2 be diagonal, then J can be written as

$$J = \frac{1}{2}\sum_{i=1}^{6} q_{1i}\left(\bar{\delta}_{k+1}-\bar{\delta}_k\right)_i^2 + (1-s)\sum_{i=1}^{6} q_{2i}\nu_i\left(\bar{\delta}_{k+1}-\bar{\delta}_k\right)_i$$

Let us now define $\gamma_{ki} \equiv f_i\bar{\delta}_{ki}$, where the f_i's are chosen such that the dynamic ranges of all the γ_{ki}'s are equal. For the Skylab CMGs we can use $f_i = 1$, $i = 1, 3, 5$; and $f_i = .5$, $i = 2, 4, 6$; since the outer gimbal angles have a range twice that of the inner gimbal angles. Finally, choose $\nu_i = \delta_{ki}^n$ where n is an odd integer. With these definitions, J becomes

$$J = \frac{1}{2}\sum_{i=1}^{6} q_{1i}\,\frac{1}{f_i^2}\left(\gamma_{(k+1)i}-\gamma_{ki}\right)^2 + (1-s)\sum_{i=1}^{6} q_{2i}\,\frac{\gamma_{ki}^n}{f_i^{n+1}}\left(\gamma_{(k+1)i}-\gamma_{ki}\right)$$

Since all the γ_{ki}'s have the same dynamic range, we choose

$$q_{1i} = f_i^2 \quad \text{and} \quad q_{2i} = f_i^{n+1}$$

Thus we have for the Skylab

$$Q_1 = \text{diag}\left(1,\ .25,\ 1,\ .25,\ 1,\ .25\right)$$

$$Q_1^{-1}Q_2 = \text{diag}\left(1,\ .5^{n-1},\ 1,\ .5^{n-1},\ 1,\ .5^{n-1}\right)$$

and

$$\nu' = \left(\bar{\delta}^n_{k1}, \bar{\delta}^n_{k2}, \ldots, \bar{\delta}^n_{k6}\right)$$

By choosing n = 5 (the value of the exponent used in the MSFC rotation law) and (1-s) = .2, we obtain, using Eq. (10a), gimbal angle trajectories nearly identical with those of the MSFC control law.

Thus, the algorithms derived here, using the torque matching approach, can duplicate the results using the MSFC law. Whether this approach should be used instead of the MSFC approach depends on their relative ease of implementation. This is discussed in the concluding remarks.

These case studies suggest that at least some rate control (i.e., s > 0) is essential to obtain smoothly evolving gimbal angles.

B. "Momentum Matching" at the Sampling Points

In the method just described, we matched the actual torque, $T^a = \dot{H} + \omega \times H$, with the desired torque T^d at the sampling times. Here we will attempt to satisfy the integrated form of the constraint (Eq. (2)), and work with the momentum vector $H(\delta(t))$ instead.

Given δ_k, ω_k, T^d_k, we seek a sequence of gimbal angles δ_{k+1} which minimizes a cost function $J(\delta_{k+1}, \delta_k)$, subject to the constraint

$$H\left(\delta_{k+1}\right) = \sum_{j=0}^{k} T^d_j \Delta t - \sum_{j=0}^{k} \omega_j \times H\left(\delta_j\right)\Delta t \qquad (12)$$

We assume that for $k = 0$, δ_0 is defined and satisfies suitable pre-determined properties.

As in the first approach, we can, in this fashion, generate a sequence $\{\delta_k\}$, and, therefore, a sequence $\{\dot{\delta}_k\}$ by

$$\dot{\delta}_k = \frac{\delta_{k+1} - \delta_k}{\Delta t}$$

Similarly, we define the function $\delta(t)$ on the interval $[t_k, t_{k+1}]$ by

$$\delta(t) = \dot{\delta}_k + \delta_k\left(t - t_k\right)$$

Should the gimbal angles, δ, evolve according to this trajectory, the actual torque, T^a, produced in the interval $[t_k, t_{k+1}]$ would be, according to Eq. (1)

$$T^a = C\left(\delta(t)\right)\dot{\delta}(t) + \omega(t) \times H\left(\delta(t)\right) = \dot{H} + \omega \times H$$

Thus,

$$\int_0^t T^a d\tau = H\left(\delta(t)\right) + \int_0^t \omega(\tau) \times H\left(\delta(\tau)\right) d\tau - H\left(\delta_0\right)$$

The total error of T^a over the interval $[t_k, t_{k+1}]$, then, is

$$\int_{t_k}^{t_{k+1}} \left(T^a - T^d\right) d\tau = H\left(\delta_{k+1}\right) - H\left(\delta_k\right) + \int_{t_k}^{t_{k+1}} \omega \times H d\tau - T_k^d \Delta t$$

From Eq. (12), we have

$$H\left(\delta_{k+1}\right) - H\left(\delta_k\right) = T_k^d \Delta t - \omega_k \times H_k \Delta t$$

This yields the error

$$\frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \left(T^a - T^d\right) d\tau = \frac{1}{\Delta t}\int_{t_k}^{t_{k+1}} \left(\omega \times H - \omega_k \times H_k\right) d\tau$$

This is precisely the second term of the torque error obtained in the previous approach (Eq. (6)).

An advantage of this approach is now readily apparent: If the spacecraft is held in an inertial attitude, which is indeed what is desired in the Skylab, then $\omega = 0$ for all t. This means that the total error over $[t_k, t_{k+1}]$, and, therefore, the total error up to any sampling time t_k, is zero. Theoretically, at least, then this control approach, for $\omega = 0$, produces a torque which, on average, is the same as the desired torque up to any time t_k. This compares with the possibility of accumulating errors as in the first approach.

On the other hand, in the case of a quadratic cost function $J(\delta_{k+1}, \delta_k)$, this approach is not as simple to execute as is the first. In the previous case, given the angle vector δ_k, a one-time application of Eq. (10) yields the next vector δ_{k+1}. In the present case, however, iterations may be required.

Again, for the sake of brevity, let

$$\bar{\delta}_k = \delta_k - \hat{\delta}$$

for some constant reference vector $\hat{\delta}$. Let J be the same quadratic cost function as before. We seek to minimize

$$J = \frac{1}{2}\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right)' Q\left(\delta_k\right)\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right) \qquad (8)$$

subject to constraint

$$H\left(\delta_{k+1}\right) = \sum_{j=0}^{k}\left(T_j^d - \omega_j \times H\left(\delta_j\right)\right)\Delta t \equiv \sum_{j=0}^{k} T_j^c \Delta t \qquad (12)$$

An application of the Lagrange multiplier method yields a system of 9 equations in 9 unknowns:

$$\begin{aligned} H\left(\delta_{k+1}\right) - \sum_{j=0}^{k} T_j^c \Delta t &= 0 \\ Q\left(\delta_k\right)\left(\bar{\delta}_{k+1} - s\bar{\delta}_k\right) + C_k' \lambda &= 0 \end{aligned} \qquad (13)$$

where λ is the 3×1 vector of Lagrange multipliers.

We shall solve Eq. (13) for δ_{k+1} by Newton's method of iterations. [A brief recapitulation of that method is given in Appendix 2.] Newton's method generally is unwieldy for higher dimensional systems, primarily because it requires the inversion of the Jacobian matrix, M, which in our case is 9×9:

$$M = \begin{pmatrix} C_{3\times 6} & 0_{3\times 3} \\ \hline Q+D_{6\times 6} & C'_{6\times 3} \end{pmatrix}$$

where D is the 6×6 matrix with entries d_{ij}, which are partial derivatives of the $i^{\underline{th}}$ component of the vector $C'(\delta)\lambda$ with respect to the $j^{\underline{th}}$ component of δ. However, in this case, Newton's method is easily executed.

First, because of the simple block form for M, M^{-1} can be obtained analytically, whenever it exists, as

$$M^{-1} = \left(\begin{array}{c|c} (Q+D)^{-1}C'\left(C(Q+D)^{-1}C'\right)^{-1}_{6\times 3} & (Q+D)^{-1} - (Q+D)^{-1}C'\left(C(Q+D)^{-1}C'\right)^{-1}C(Q+D)^{-1}_{6\times 6} \\ \hline -\left(C(Q+D)^{-1}C'\right)^{-1}_{3\times 3} & \left(C(Q+D)^{-1}C'\right)^{-1}C(Q+D)^{-1}_{3\times 6} \end{array}\right) \tag{14}$$

Second, because of the special nature of the matrix $C(\delta)$ (where the first two components of δ appear only in the first two columns of C, the third and fourth components of δ appear only in the third and fourth columns of C, and so on), the matrix $Q + D$ has the simple block form

$$Q + D = \begin{pmatrix} G_1 & 0 & 0 \\ 0 & G_2 & 0 \\ 0 & 0 & G_3 \end{pmatrix}$$

where each G_i is 2×2. Therefore, $(Q+D)^{-1}$ is simply

$$(Q+D)^{-1} = \begin{pmatrix} G_1^{-1} & 0 & 0 \\ 0 & G_2^{-1} & 0 \\ 0 & 0 & G_3^{-1} \end{pmatrix}$$

whenever the G_i^{-1} exist.

Because of these simplifying properties, it is possible to obtain an explicit analytic formula for the iterative scheme. In fact, let $\bar{\gamma}_j$ be the $j^{\underline{th}}$ approximation for the angle $\bar{\delta}_{k+1}$ being sought. Then, an application of Eq. (A2.2) of Appendix 2 together with Eq. (14), yields

$$\begin{aligned} \bar{\gamma}_{j+1} - \bar{\gamma}_j = & \left(Q(\gamma_j) + D(\gamma_j)\right)^{-1}C'(\gamma_j)\left(C_j(Q_j+D_j)^{-1}C_j'\right)^{-1}\left(\sum_{m=0}^{k} T_m^C \Delta t - H(\gamma_j)\right) \\ & + (Q_j+D_j)^{-1}\left[C_j'\left(C_j(Q_j+D_j)^{-1}C_j'\right)^{-1}C_j(Q_j+D_j)^{-1} - I\right]Q_j\left(\bar{\gamma}_j - s\bar{\delta}_k\right) \end{aligned} \tag{15}$$

$$\lambda_{j+1} = -\left(C_j\left(Q_j+D_j\right)^{-1}C_j'\right)^{-1}C_j\left(Q_j+D_j\right)^{-1}\left(\bar{\gamma}_j - s\bar{\delta}_k\right)$$

$$- \left(C_j\left(Q_j+D_j\right)^{-1}C_j'\right)^{-1}\left(\sum_{m=0}^{k} T_m^c\Delta t - H_j\right) \qquad (15)$$

Thus, the difficulties which are generally associated with Newton's method for solving higher dimensional systems disappear in the present case.

We observe here that Eqs. (15) actually contain, as a special case, Eq. (10) used in the previous approach if we set the matrix D equal to zero, set $\bar{\gamma}_0 = \bar{\delta}_k$, and perform only one iteration to obtain each $\bar{\delta}_{k+1}$. In that case, the first equation of Eqs. (15) is precisely of the form of Eq. (10), if we recognize that

$$\sum_{m=0}^{k} T_m^c\Delta t - H\left(\gamma_0\right) = \sum_{m=0}^{k} T_m^c\Delta t - H\left(\delta_k\right)$$

$$= \sum_{m=0}^{k} T_m^c\Delta t - \sum_{m=0}^{k-1} T_m^c\Delta t = T_k^c\Delta t$$

Setting D = 0 is equivalent to assuming that the function H varies linearly in each sampling interval. Thus, in that sense, the first approach is a linearized version of the present one.

Equation (15) is valid whenever $(Q+D)^{-1}$ and $(C(Q+D)^{-1}C')^{-1}$ exist. If $(Q+D)^{-1}$ exists, $(C(Q+D)^{-1}C')^{-1}$ exists if and only if the three gyro momentum vectors are not colinear (See Appendix 1.) If (Q+D) is singular at some sample time t_k, then so is the Jacobian matrix M, and Newton's method fails at that point. In the actual program that was used, whenever this happened, we merely employed Eq. (10) to obtain the next δ_{k+1}.

In the case where $\omega = 0$, $s = 0$, and Q = constant matrix, this approach also possesses, theoretically at least, the desirable property that the gimbal angles remain fixed when the

desired torque becomes zero at a certain time t_N and stays zero for all time thereafter. For, if $\omega = 0$, and if $T_k^d = 0$ for all $k > N$, then the constraint Eq. (12) for δ_{N+1} becomes

$$H\left(\delta_{N+1}\right) = \sum_{j=0}^{N-1} T_j^d \Delta t$$

Since $s = 0$, the minimizing problem given by Eqs. (8) and (12) becomes: find δ_{N+1} which minimizes

$$J = \bar{\delta}'_{N+1} \, Q\bar{\delta}_{N+1}$$

such that

$$H\left(\delta_{N+1}\right) = \sum_{j=0}^{N-1} T_j^d \Delta t$$

But this is the same problem as the one which defines δ_N. Hence, if we use δ_N as a first guess to find δ_{N+1}, we must have $\delta_{N+1} = \delta_N$. In fact, we must have, by repetition of this argument, $\delta_k = \delta_N$ for all $k > N$.

We now discuss some typical case studies using this approach. Computer runs were made with a program based on Eqs. (15) and (10). That is, given a gimbal angle vector δ_k and a desired torque T_k^d, we generate a new angle vector using the simpler routine Eq. (10). If for this new angle vector the momentum constraint Eq. (12) is satisfied, then this vector is our desired solution δ_{k+1}. If not, then iterations are performed using Eq. (15), with the angle vector generated in the first step by Eq. (10) being a first "guess." The use of Eq. (10) to provide the first approximation, thus, has the advantage of reducing computation time. It also has the added advantage of continuing in time if at a certain time t_k, the Jacobian matrix becomes singular, and Newton's method fails.

In the results presented here, we have considered only the "open-loop" case, the driving torque being the gravity-gradient torque. The matrix Q is taken to be the identity matrix, and we have assumed an inertially fixed attitude - that is, $\omega = 0$. That last assumption yields the fact that the average torque error over the interval $[t_k, t_{k+1}]$, and, therefore, the total error up to any sampling time, is always zero. This fact is borne out in the actual computations.

Various runs with different values of s and Δt were made. One immediate observation is that the resulting gimbal angle trajectories are extremely insensitive to changes in the rate mixture s and the sampling interval Δt. It is not until $s = .99$ and beyond that there are any identifiable changes in the plots of the angles. Another fact to be observed is that, most of the time, iterations are not required; that is, Eq. (10) provides very accurate computation most of the time, using occasional iterations to eliminate accumulated errors. The required iterations decrease even further with s values nearing 1, so that computation times, for a fixed Δt and for these values of s, become comparable to that of the first approach. (The computation times for fixed s changes of course, almost linearly with Δt.)

Figure 8 compares results for $s = .75$, and $s = .995$ with $\Delta t = .2$ sec. Notice the similarity in results between the $s = .75$ case and the same case using the first approach. Notice also that for the case $s = .995$ the jumps near 4100 sec are smoothed out.

This approach provides, in general, smoother response for every value of s than the first because of its increased accuracy. This is at the cost of a slightly longer program and longer computation time due to occasional iterations. However, the computation times become comparable as s nears 1. Because of the extreme insensitivity of the results to changes in both Δt and s, one may obtain smooth response with a short computation time by using larger values of s and Δt.

III. Concluding Remarks

We have presented here a unified treatment of CMG gimbal angle control in the framework of an optimization problem. Two approaches for control have been given, each with its own relative merits, and some of their properties discussed. Case studies using quadratic penalty

functions were investigated. These studies demonstrated the need for including a rate term in the penalty function in order to achieve smoothly evolving gimbal angle trajectories.

By a proper choice of the penalty function, the torque-matching approach appears to duplicate the gimbal angle trajectory obtained using the MSFC law.

IBM, Huntsville, compared these two control laws with respect to memory requirements and execution time (Ref. 3). They assumed correctly that the supporting logic for each law would be about the same. The results of the assessment are summarized as follows:

	Control Law Proper		Control Law plus Logic	
	Memory Locations	Time	Memory Locations	Time
MSFC Law	755	31.5 ms	1502	68.0 ms
Torque Matching	531	48.3 ms	1278	85.3 ms

A memory saving of 224 words could be realized. The execution time, however, would increase 17.3 ms; this represents an increase in the control law duty cycle from 34.0% to 42.7%. But this duty cycle is based on executing the CMG control law 5 times per second, a requirement that could be reduced. Consequently, if the need for additional memory becomes compelling, the torque matching law should be considered for replacing the present MSFC law.

Acknowledgement

We express our sincere gratitude to R. W. Grützner for his patience and his expert and competent programming of these control laws. Thanks are also due to D. A. DeGraaf for many illuminating discussions and to W. B. Chubb of MSFC for having IMB assess the first control law presented here.

S. C. Chu

J. Kranton

1032-$^{SCC}_{JK}$-jct

Attachments:
Figures 1 thru 8

REFERENCES

1. Kranton, J., "Application of Optimal Control Theory to Attitude Control with Control Moment Gyroscopes," D. Sc. dissertation, February 1970, George Washington University, Washington, D. C.

2. "Apollo Telescope Mount Digital Computer Program Definition Document," Part I, Section 12, November 4, 1970, IBM Federal Systems Division, Huntsville, Ala.

3. McPherson, B. W., "Results of the Memory Assessment of the Bellcomm CMG Control Law Proposal," Memorandum, IBM Federal Systems Division, Huntsville, Ala., April 8, 1971.

APPENDIX 1

PROPERTIES OF THE MATRIX $CQ^{-1}C'$

The validity of Eqs. (10) and (15) depend on the non-singularity of the matrices Q, $CQ^{-1}C'$, and (Q+D), $C(Q+D)^{-1}C'$. The non-singularity of Q is assumed from the outset; in fact, we assume Q is positive definite. The matrix Q+D varies during the Newton iterative process and not much can be said about it a priori. If Q+D becomes singular, then the Jacobian matrix M becomes singular, and Newton's method fails.

We prove here conditions under which $CQ^{-1}C'$ becomes singular.

Theorem 1. If Q is positive definite, then $CQ^{-1}C'$ is non-singular if and only if C is of rank 3.

Proof: Suppose C is of rank 3. Since Q is positive definite, so is Q^{-1}. Let the row vectors of C be denoted by C^i, i = 1, 2, 3. Suppose, to the contrary, that $CQ^{-1}C'$ is singular. Then there exists a non-zero vector $X' = (x_1, x_2, x_3)$ such that

$$X'CQ^{-1}C' = 0 \qquad \text{(A1.1)}$$

Let $Y = X'C$. Then (A1.1) implies that Y is orthogonal to each of the vectors $Q^{-1}C^{i'}$. Since C is of rank 3, the row vectors of C and, therefore, the column vectors of C' are linearly independent. Q^{-1} is non-singular; thus, the vectors $Q^{-1}C^{i'}$ are linearly independent. Let the space spanned by these vectors, $Q^{-1}C^{i'}$, be denoted by S. Then $Y \notin S$.

On the other hand,

$$Q^{-1}Y = Q^{-1}C'X = Q^{-1}\left(x_1C^{1'} + x_2C^{2'}\ x_3C^{3'}\right)$$

$$= x_1Q^{-1}C^{1'} + x_2Q^{-1}C^{2'} + x_3Q^{-1}C^{3'}$$

Thus, $Q^{-1}Y$ is a linear combination of the $Q^{-1}C^{i'}$ and must be in the space S. Therefore, Y and $Q^{-1}Y$ are in complementary spaces, and we must have

$$Y'Q^{-1}Y = 0$$

But, X is non-zero, C is of rank 3; thus $Y' = X'C$ is also non-zero. This contradicts the assumption that Q is positive definite. Thus, if C is of rank 3, $CQ^{-1}C'$ is non-singular.

On the other hand, if C is of rank less than 3, then there exists a non-zero vector $X = (x_1, x_2, x_3)$ such that $C'X = 0$. Therefore, $CQ^{-1}C'X = 0$ and $CQ^{-1}C'$ is singular. This completes the proof of the theorem.

Theorem 2. If Q is positive definite, $CQ^{-1}C'$ is non-singular if and only if the momentum vectors, h_i, of the individual gyros are all colinear.

Proof: From Theorem 1, $CQ^{-1}C'$ is non-singular if and only if C is of rank 3. C is of rank 3 if and only if 3, and exactly 3, of its column vectors are linearly independent. Thus C is of rank less than 3 if and only if all of its column vectors are co-planar. We shall now derive the conditions of co-planarity.

Denote by V_i the column vectors of C. Observe first that $V_1 \perp V_2$, $V_3 \perp V_4$, $V_5 \perp V_6$. The planes determined by V_1 and V_2, V_3 and V_4, and V_5 and V_6 are, respectively:

$$(x,y,z) \cdot V_1 \times V_2 = 0 \qquad \text{(A1.2)}$$

$$(x,y,z) \cdot V_3 \times V_4 = 0 \qquad \text{(A1.3)}$$

$$(x,y,z) \cdot V_5 \times V_6 = 0 \qquad \text{(A1.4)}$$

where

$$(V_1 \times V_2)' = (-\cos\alpha_1\cos^2\beta_1,\ \sin\alpha_1\cos^2\beta_1,\ \sin\beta_1\cos\beta_1)$$

$$(V_3 \times V_4)' = (\sin\beta_2\cos\beta_2,\ -\cos\alpha_2\cos^2\beta_2,\ \sin\alpha_2\cos^2\beta_2)$$

$$(V_5 \times V_6)' = (\sin\alpha_3\cos^2\beta_3,\ \sin\beta_3\cos\beta_3,\ -\cos\alpha_3\cos^2\beta_3).$$

If all the V_i are co-planar, then the equations (A1.2) - (A1.4) represent the same plane and are, thus, equivalent equations. Therefore, the ratios of the coefficients of x, y, and z, pairwise, must be the same in all 3 equations. This fact yields:

$$\tan\beta_1 = \cos\alpha_1\,\mathrm{ctn}\alpha_3 \qquad \text{(A1.5)}$$

$$\tan\beta_2 = \cos\alpha_2\,\mathrm{ctn}\alpha_1 \qquad \text{(A1.6)}$$

$$\tan\beta_3 = \cos\alpha_3\,\mathrm{ctn}\alpha_2 \qquad \text{(A1.7)}$$

$$\tan\alpha_3 = -\mathrm{ctn}\alpha_1\,\mathrm{ctn}\alpha_2. \qquad \text{(A1.8)}$$

These four equations give the conditions for C being of rank less than 3, and, therefore, $CQ^{-1}C'$ being singular.

To complete the proof, we need now to relate these equations to the gyro momentum vectors h_i. The h_i are co-linear if and only if $h_i \times h_j = 0$ for $i \neq j$ $(i,j = 1, 2, 3)$.

$$h_1 \times h_2 = (\sin\alpha_1\sin\alpha_2\cos\beta_1\cos\beta_2 + \cos\alpha_2\sin\beta_1\cos\beta_2,$$

$$\sin\beta_1\sin\beta_2 + \cos\alpha_1\sin\alpha_2\cos\beta_1\cos\beta_2,$$

$$\cos\alpha_1\cos\alpha_2\cos\beta_1\cos\beta_2 - \sin\alpha_1\cos\beta_1\sin\beta_2).$$

Eqs. (A1.5) and (A1.8) yield:

$$\cos\alpha_2\sin\beta_1 = -\sin\alpha_1\sin\alpha_2\cos\beta_1 \qquad \text{(A1.9)}$$

Eq. (A1.6) yields:

$$\sin\alpha_1\sin\beta_2 = \cos\alpha_1\cos\alpha_2\cos\beta_2 \qquad \text{(A1.10)}$$

These last two equations give

$$\sin\beta_1\sin\beta_2 = -\cos\alpha_1\sin\alpha_2\cos\beta_1\cos\beta_2 \qquad \text{(A1.11)}$$

Eqs. (A1.9) - (A1.11), thus, imply that $h_1 \times h_2 = 0$.

In a similar fashion, use of Eqs. (A1.6) - (A1.8) yields $h_2 \times h_3 = 0$. This completes the proof.

APPENDIX 2

NEWTON'S METHOD

In this Appendix we give a brief discussion of the salient features of Newton's Method.

Let F be an n-dimensional vector function of n variables defined on some domain $\Omega \subset E_n$. Assume F is continuously differentiable on Ω. We seek a vector X, such that

$$F(X) = 0 \qquad \text{(A2.1)}$$

Suppose the vector X^1 is an approximate solution to Eq. (A2.1). We now seek a correction term ΔX^1 such that $F(X^1+\Delta X^1) = 0$. The iterative procedure is one of successively obtaining such correction terms, and is based on the use of the first order Taylor approximation.

$$0 = F(X^1+\Delta X^1) \cong F(X^1) + M(X^1)\ \Delta X^1.$$

Where M is the matrix of the first order partial derivatives of F evaluated at X^1 that is

$$M(X^1) = \left(\frac{\partial F_i}{\partial X_j}\right)_{X = X^1}$$

and is called the Jacobian matrix of F. Thus, if M is non-singular, ΔX^1 can be obtained as

$$\Delta X^1 = -M^{-1}(X^1)\ F(X^1).$$

Define $X^2 = X^1+\Delta X^1$; in a similar way, we can obtain a correction term ΔX^2. This process is continued by the formula

$$\Delta X^j = -M^{-1}(X^j)\ F(X^j) \qquad \text{(A2.2)}$$

until the sequence converges.

This process is convergent provided that the initial approximation, X^1, is sufficiently close to the solution X and the Jacobian matrix M(X) is non-singular. If M is singular, then the method fails.

APPENDIX 3

SOME STABILITY CONSIDERATIONS

We address the stability question for Eq. (10) only in the case where $Q = I$ and $\omega = 0$ - i.e., when the spacecraft is in an inertially fixed attitude. The most desirable property is to have the angles δ tend to a limiting fixed vector as $t \to \infty$. What can be proved, however, is that the angles $\bar{\delta}$ tend to a limiting magnitude which is no larger than $|\bar{\delta}_N|$, the magnitude of δ_N. This can be shown thus: for $k > N$

$$0 \leq \left(\bar{\delta}_{k+1} - \bar{\delta}_k\right)'\left(\bar{\delta}_{k+1} - \bar{\delta}_k\right) = (1-s)^2 \bar{\delta}_k' \left[C_k'\left(C_k C_k'\right)^{-1} C_k - I\right]^2 \bar{\delta}_k$$

$$= -(1-s)^2 \bar{\delta}_k' \left[C_k'\left(C_k C_k'\right)^{-1} C_k - I\right] \bar{\delta}_k$$

$$\bar{\delta}_{k+1}' \bar{\delta}_k = \bar{\delta}_k' \bar{\delta}_{k+1} = (1-s)\bar{\delta}_k' \left[C_k'\left(C_k C_k'\right)^{-1} C_k - I\right] \bar{\delta}_k + \bar{\delta}_k' \bar{\delta}_k$$

On the other hand,

$$\bar{\delta}_{k+1}' \bar{\delta}_{k+1} - \bar{\delta}_k' \bar{\delta}_k = \left(\bar{\delta}_{k+1} - \bar{\delta}_k\right)'\left(\bar{\delta}_{k+1} - \bar{\delta}_k\right) - 2\bar{\delta}_k' \bar{\delta}_k + 2\bar{\delta}_{k+1}' \bar{\delta}_k$$

$$= (1-s^2)\bar{\delta}_k' \left[C_k'\left(C_k C_k'\right)^{-1} C_k - I\right] \delta_k \leq 0$$

or,

$$\bar{\delta}_{k+1}' \bar{\delta}_{k+1} \leq \bar{\delta}_k' \bar{\delta}_k \quad \text{for all } k > N$$

Thus, the sequence $\{\bar{\delta}_k' \bar{\delta}_k\}$ is monotone non-increasing; it is also bounded from below (by zero). Bounded monotone sequences have limits; therefore, $\delta_k' \delta_k$ tend to some constant. Geometrically, this means that the vectors δ_k in six dimensional space must eventually lie on a ball of minimum radius, with center at δ.

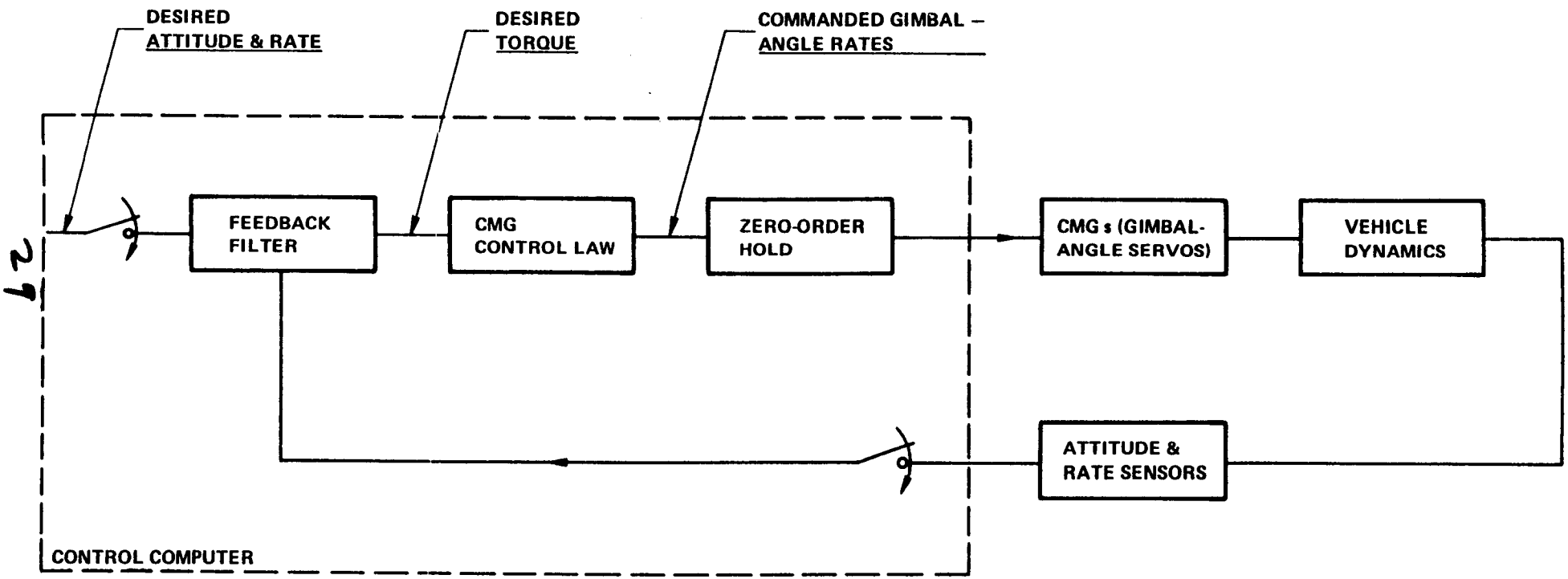
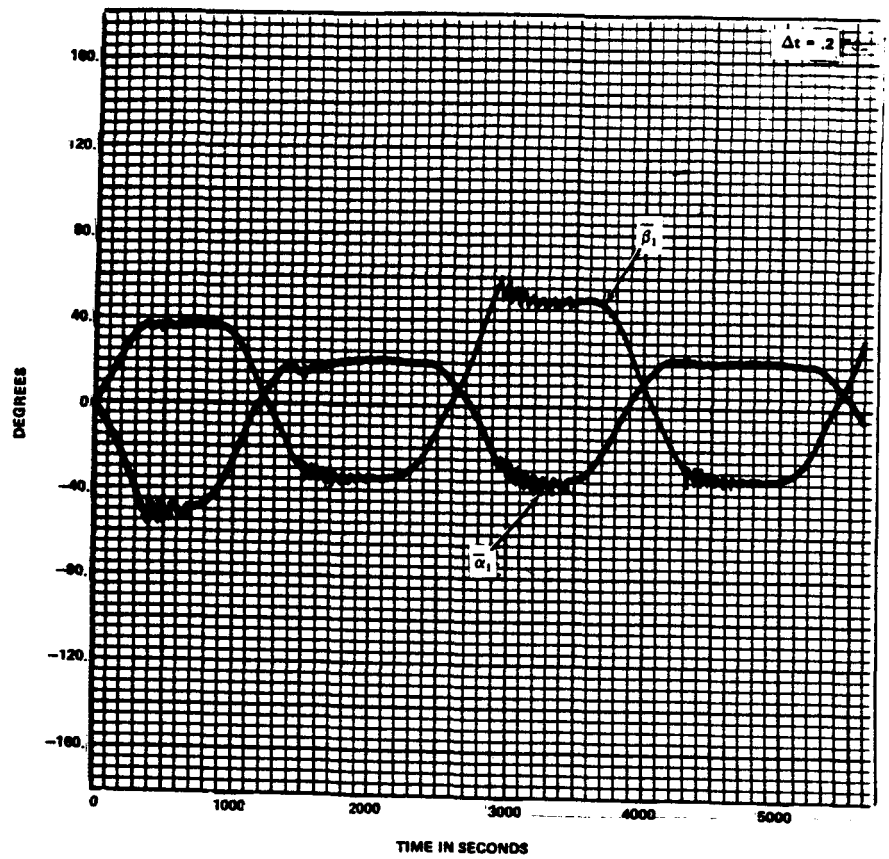
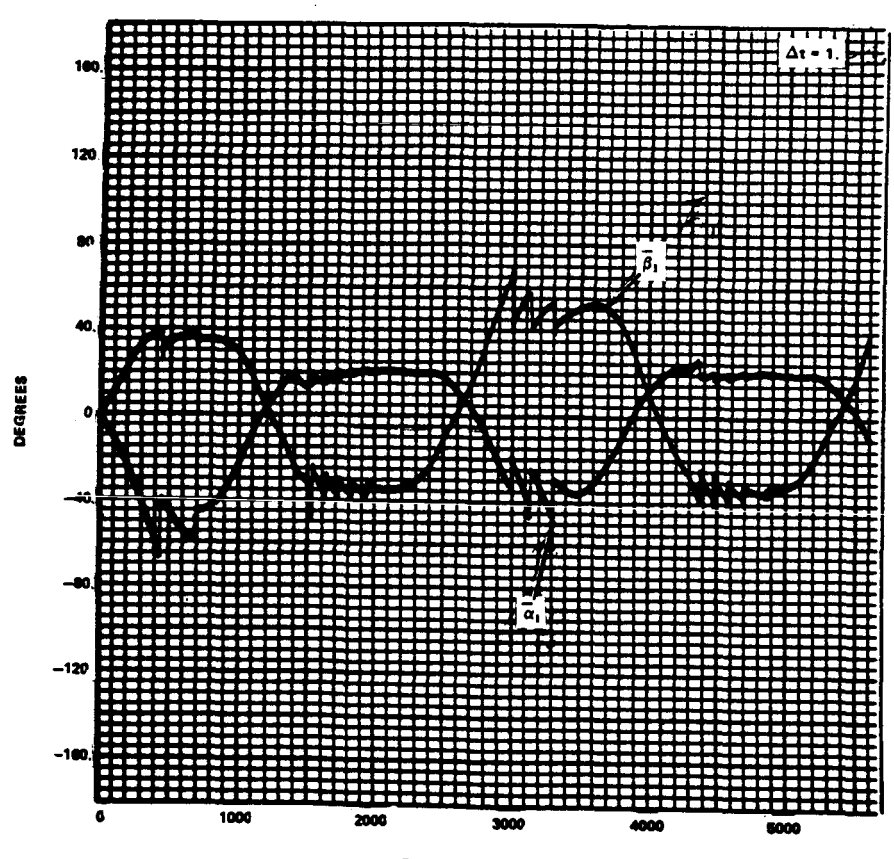


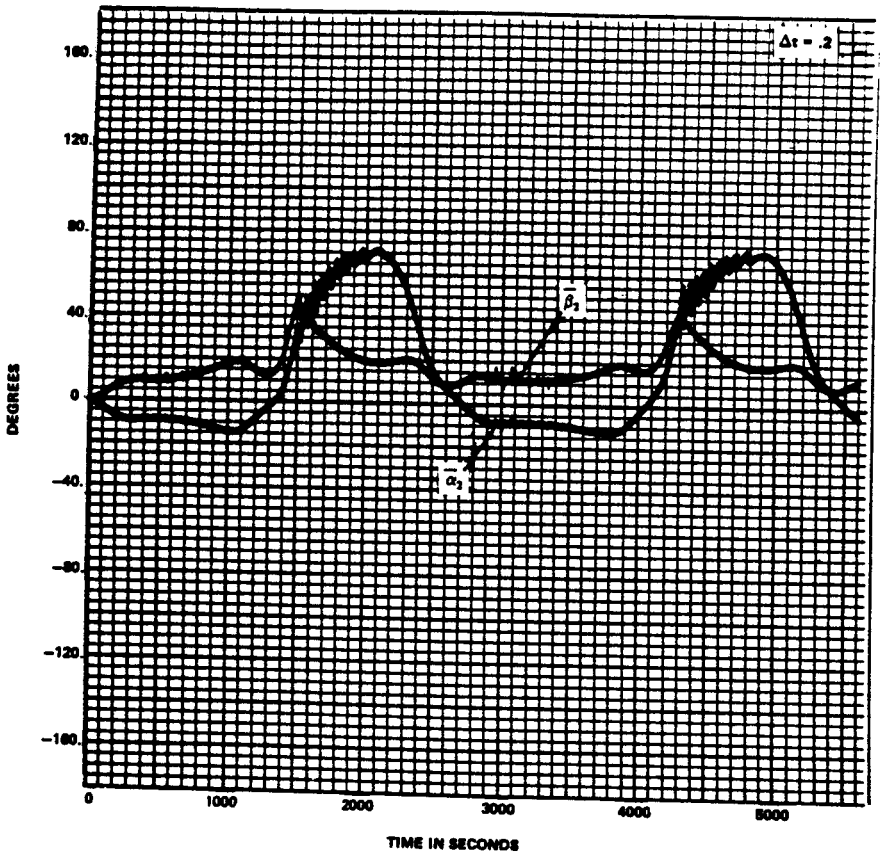
FIGURE 1 - GENERAL BLOCK DIAGRAM OF CMG CONTROL

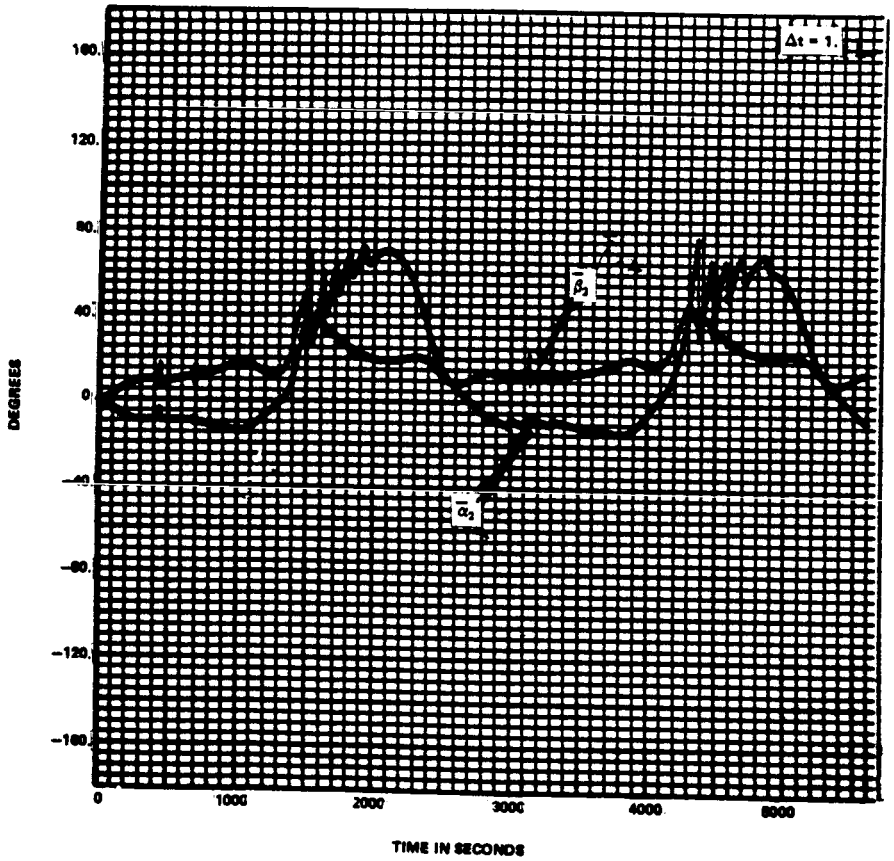




GYRO #1

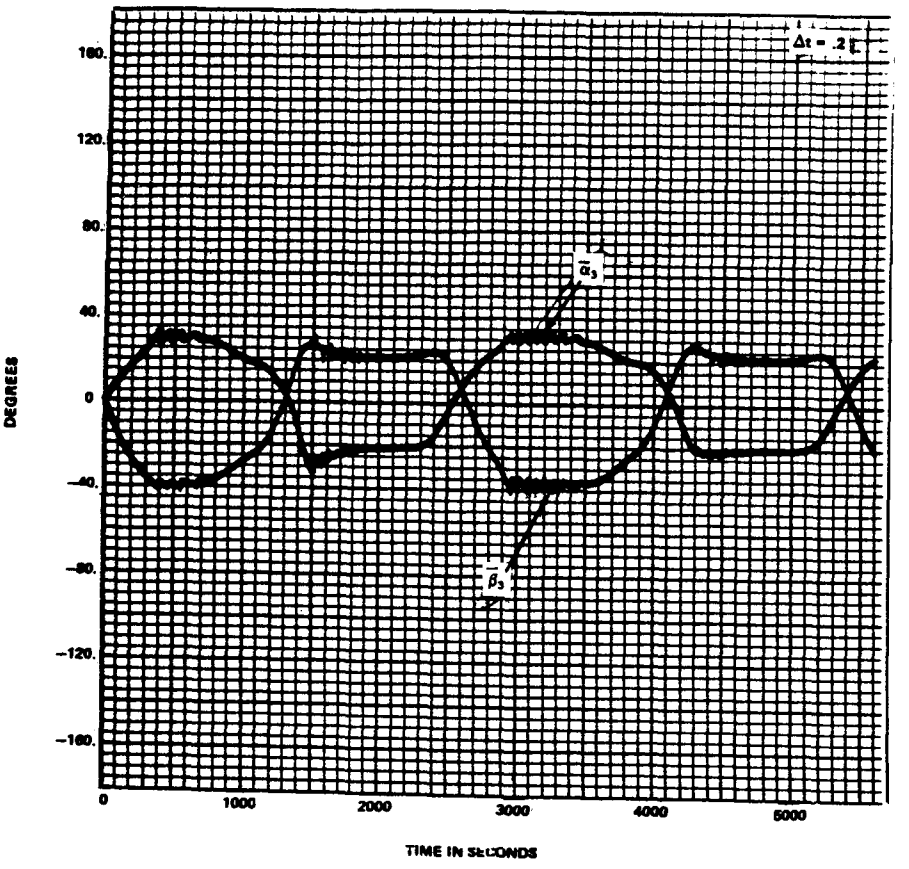
FIGURE 2A - GIMBAL ANGLES FOR ε = 0

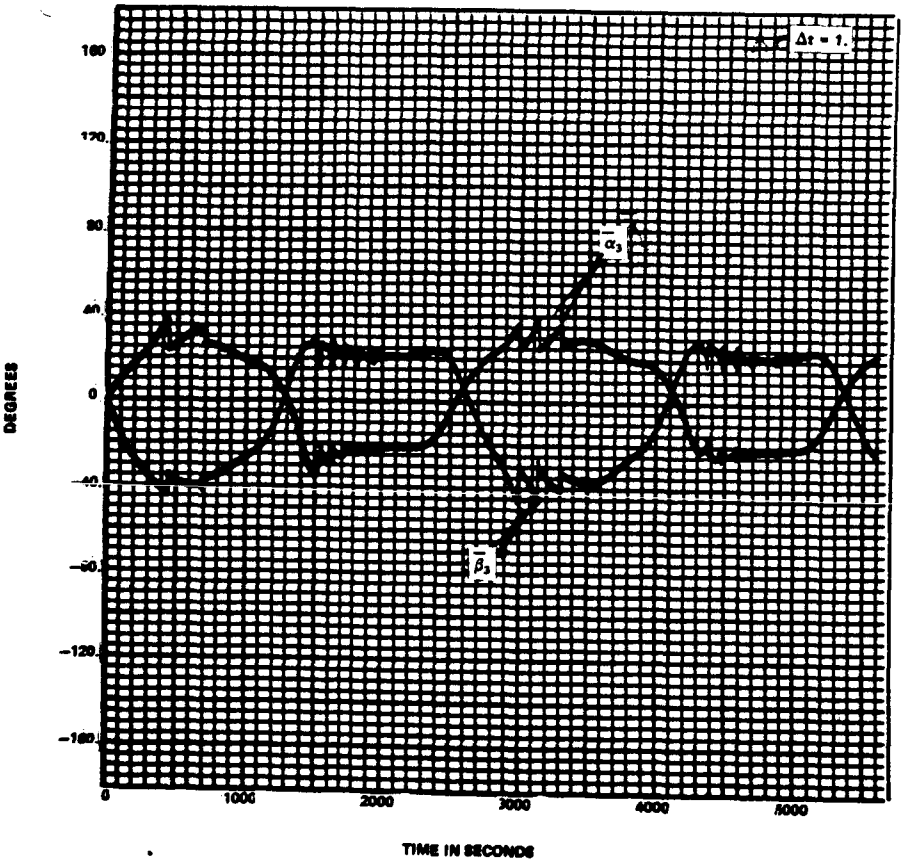




GYRO #2

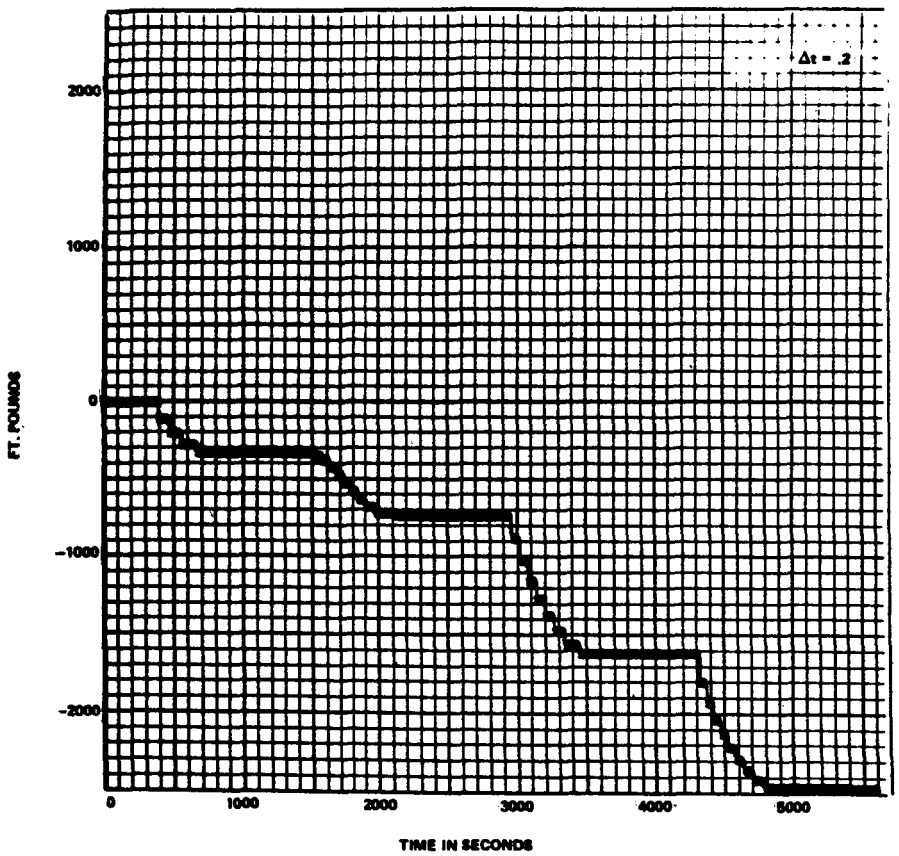
FIGURE 28 - GIMBAL ANGLES FOR $\varepsilon = 0$

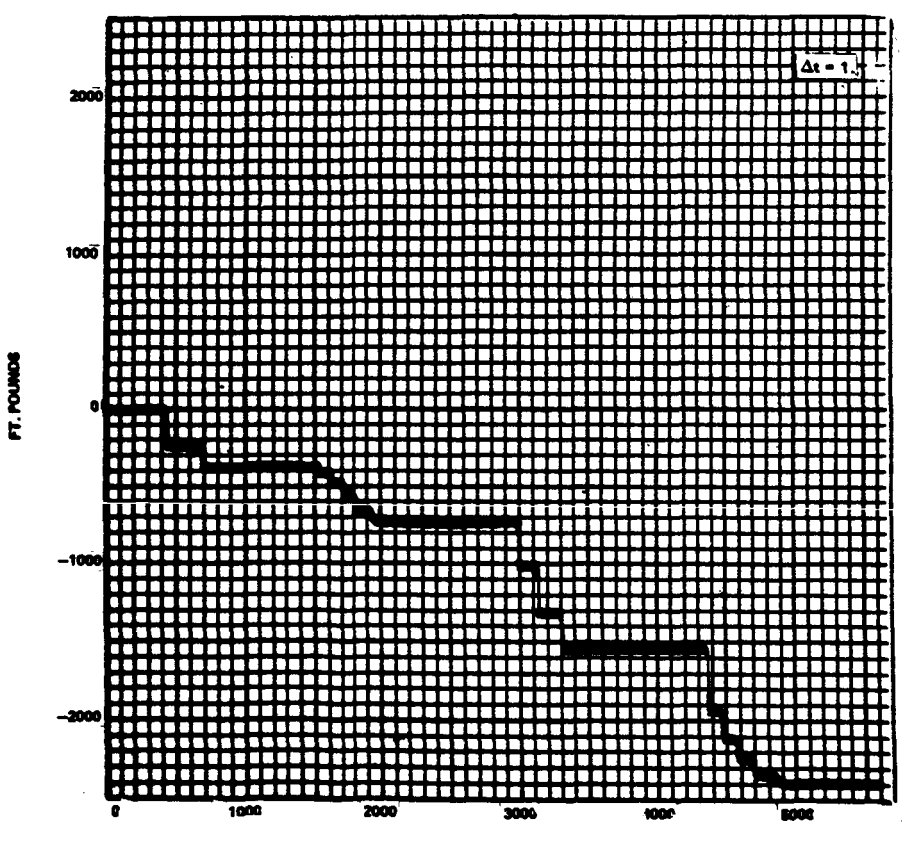




GYRO #2

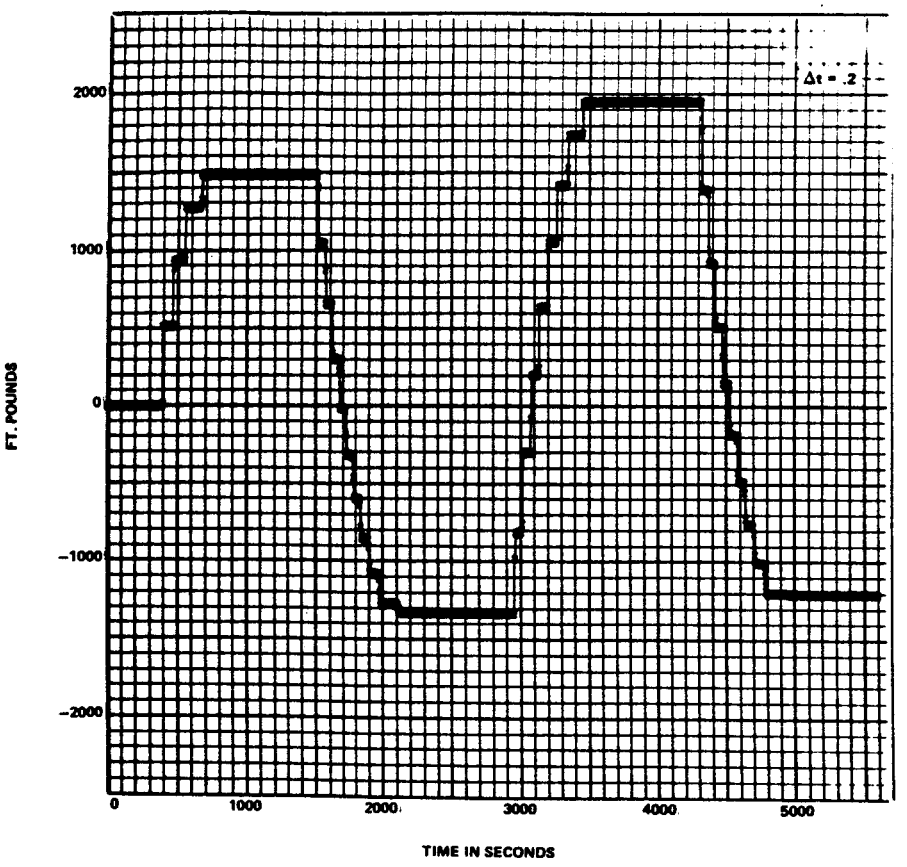
FIGURE 2C - GIMBAL ANGLES FOR $\gamma = 0$

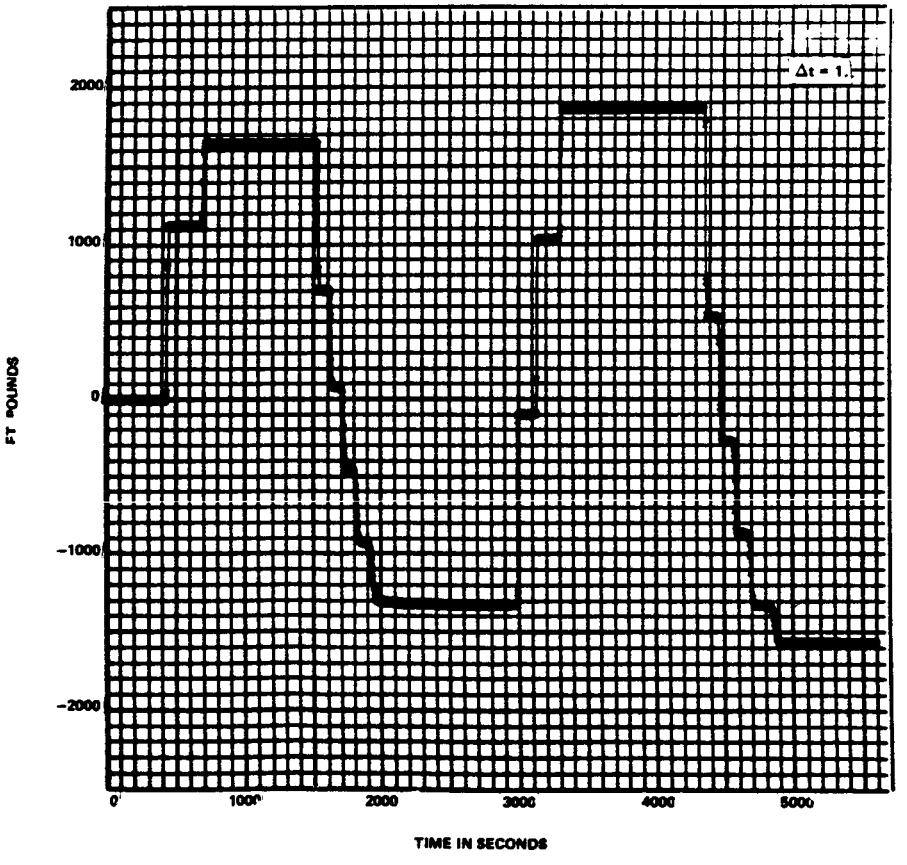




GYRO #1

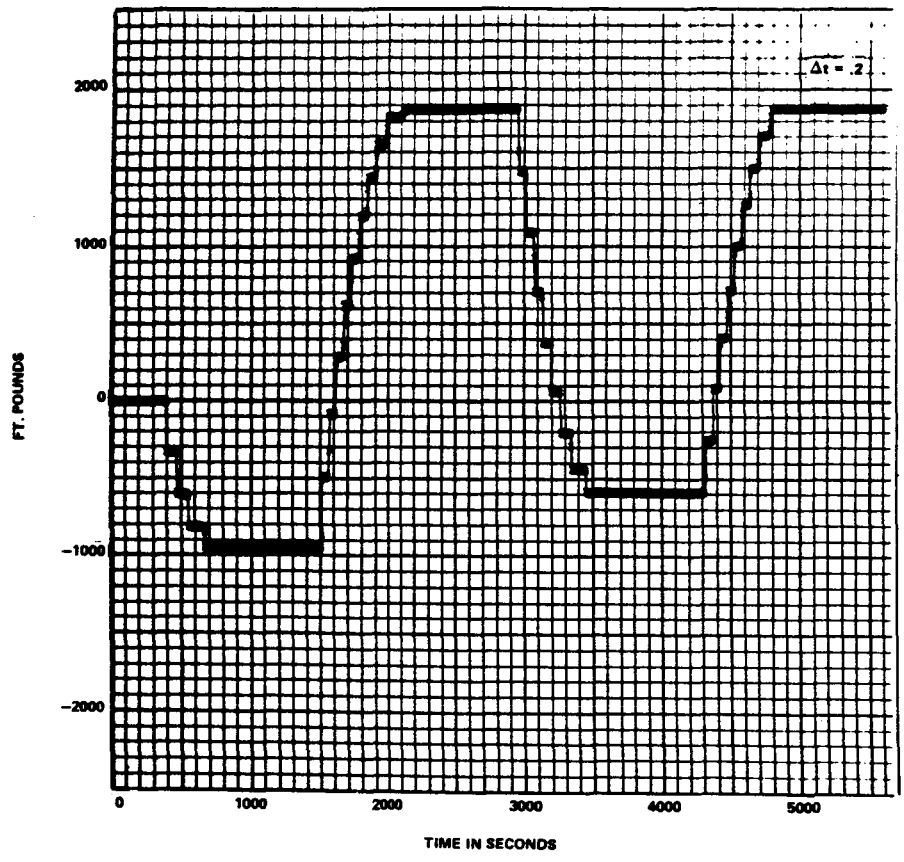
FIGURE 3A - TOTAL TORQUE ERRORS – ε = 0

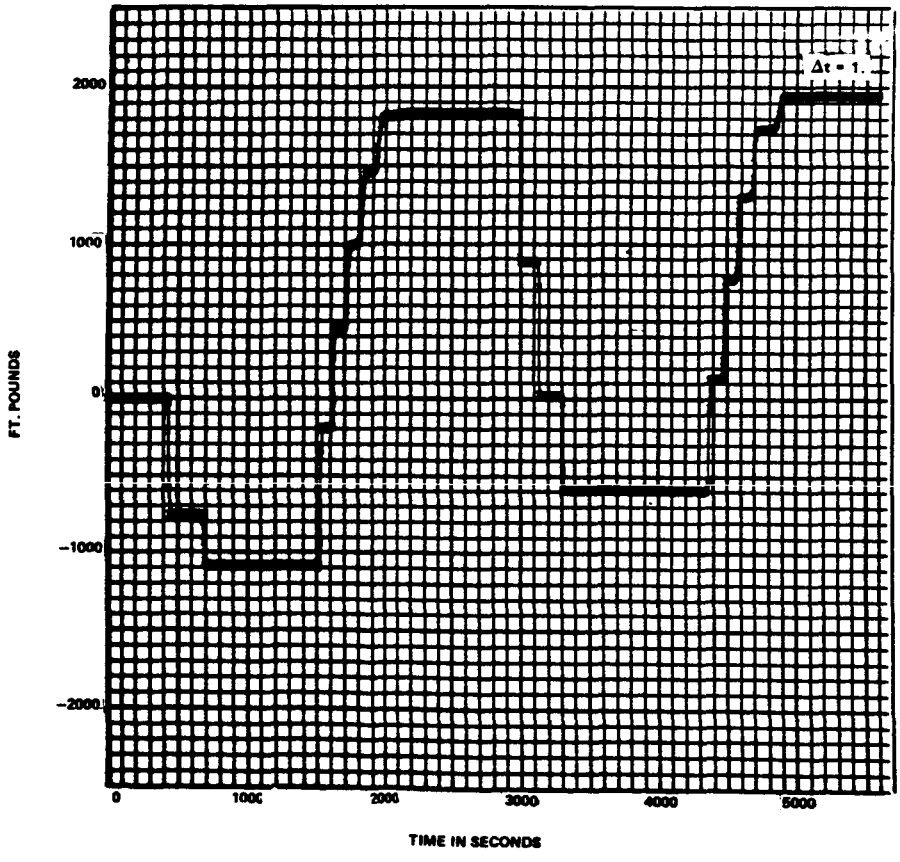




GYRO #2

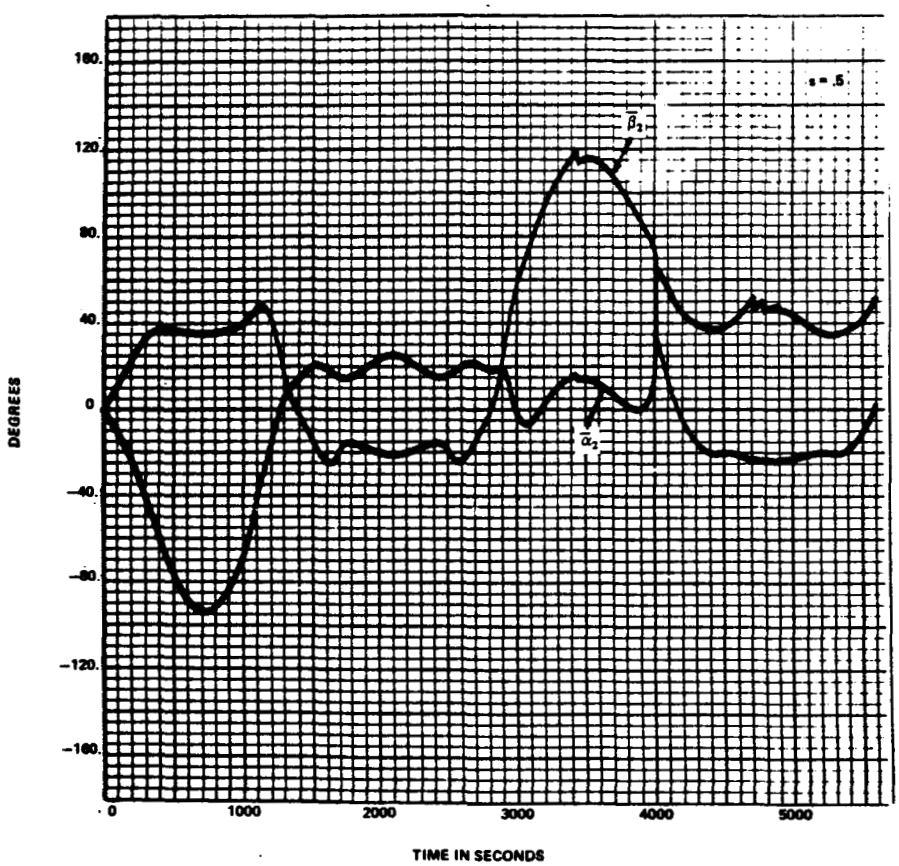
FIGURE 38 - TOTAL TORQUE ERRORS – ε = 0

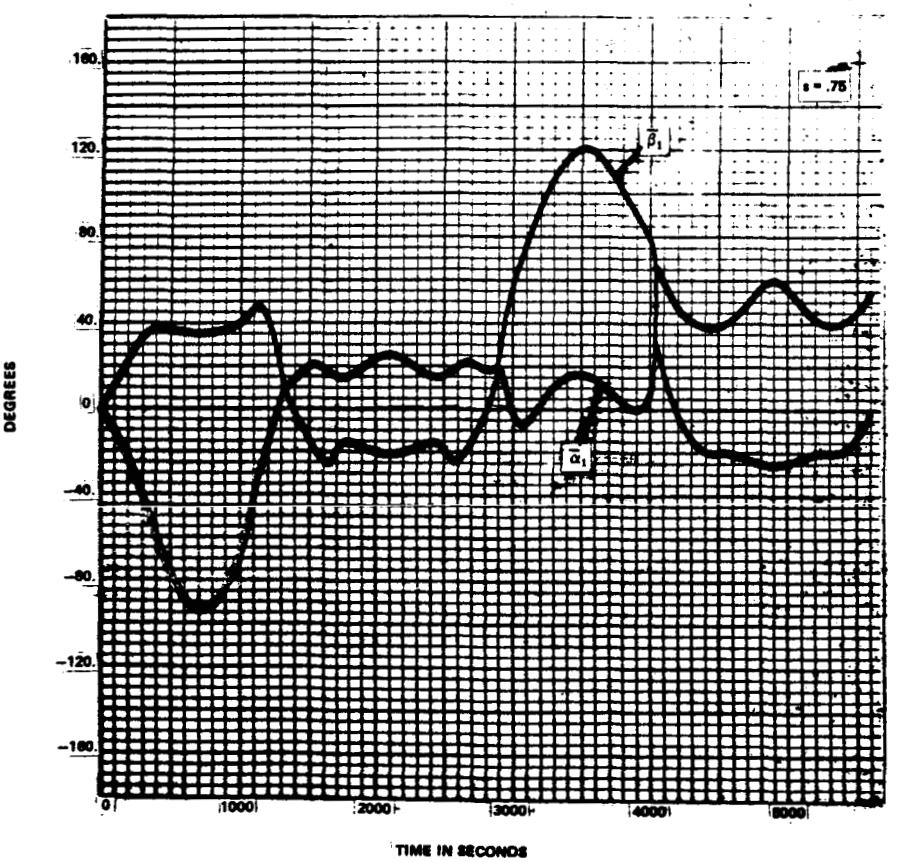




GYRO #3

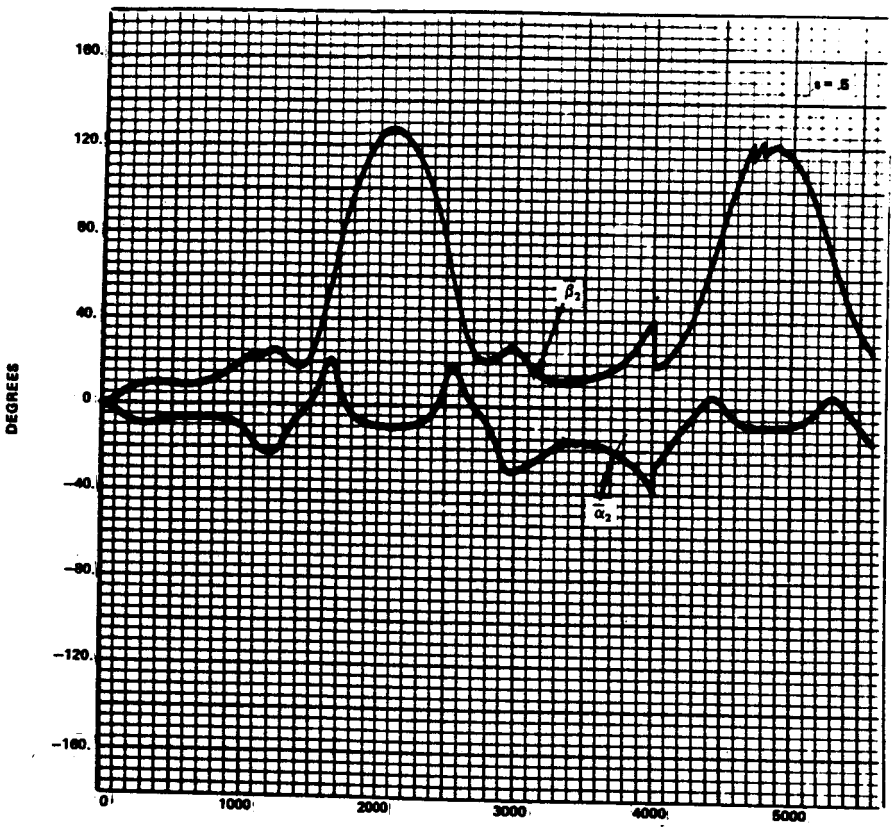
FIGURE 3C - TOTAL TORQUE ERRORS – s = 0

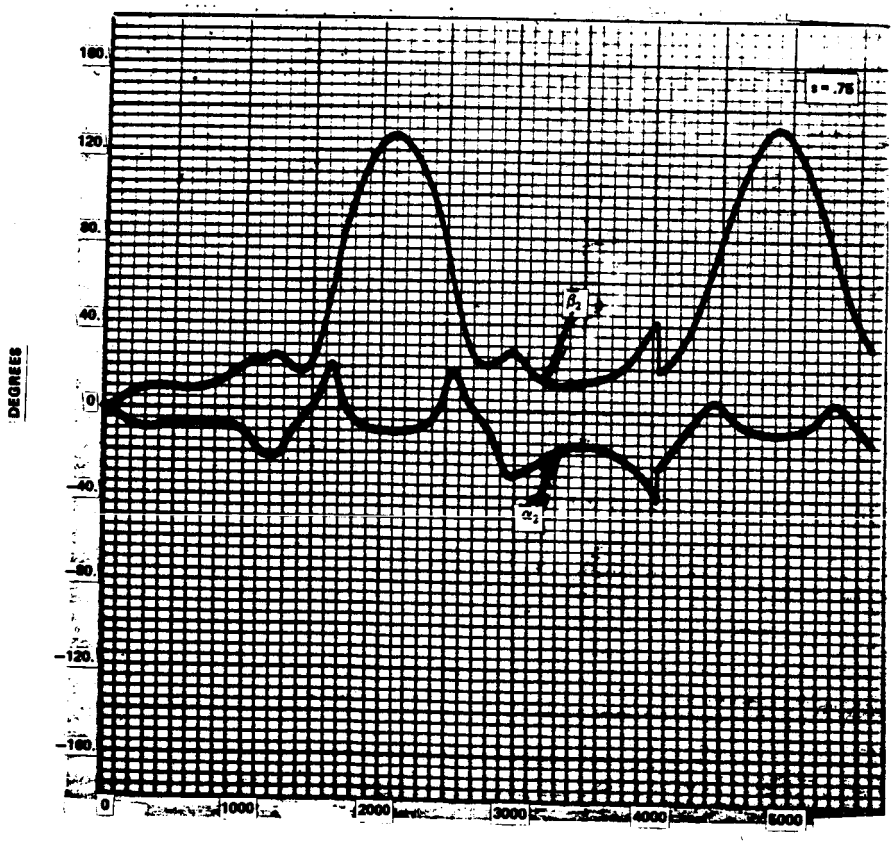




GYRO #1

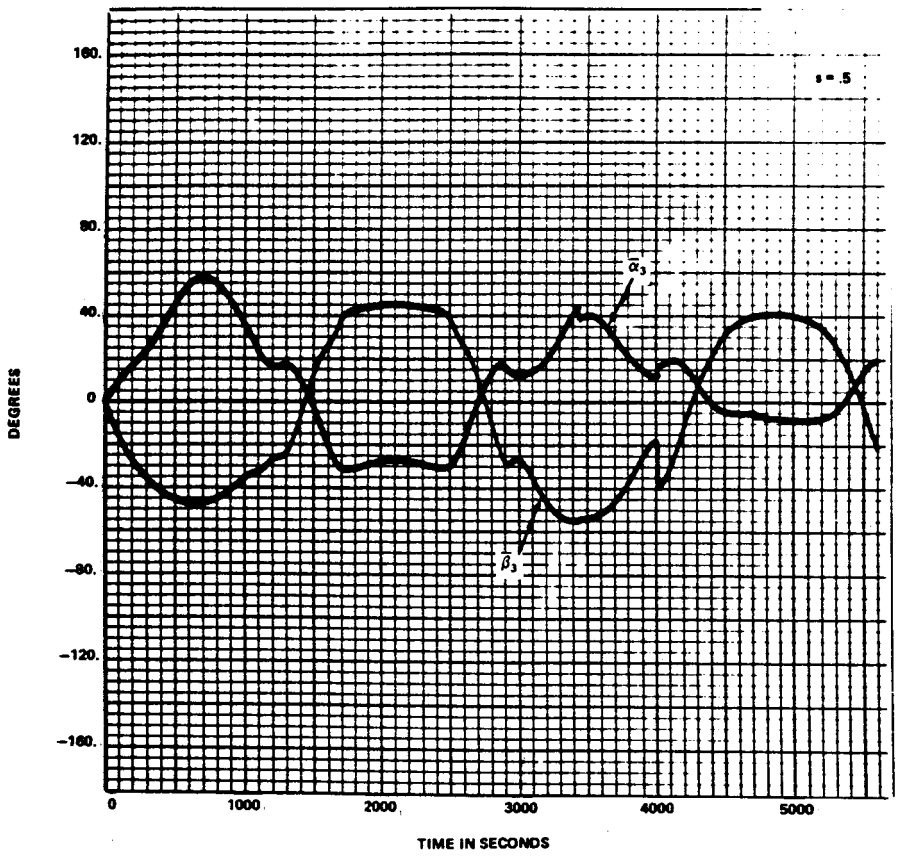
FIGURE 4A - GIMBAL ANGLES FOR $\Delta t = .2$

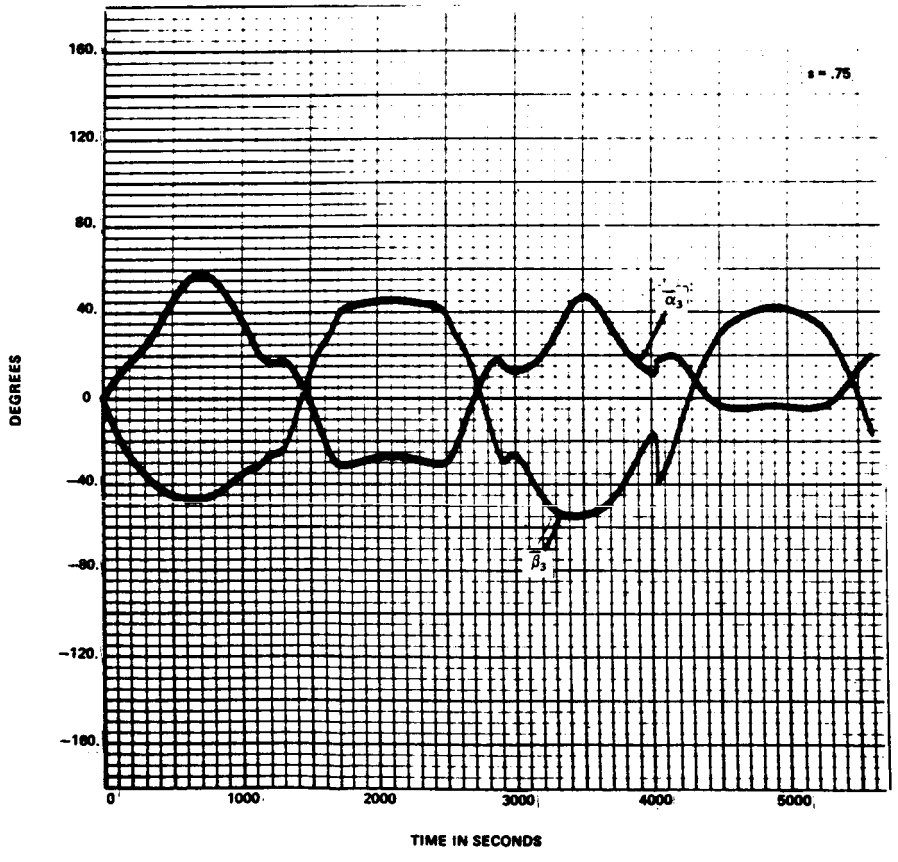




GYRO #2

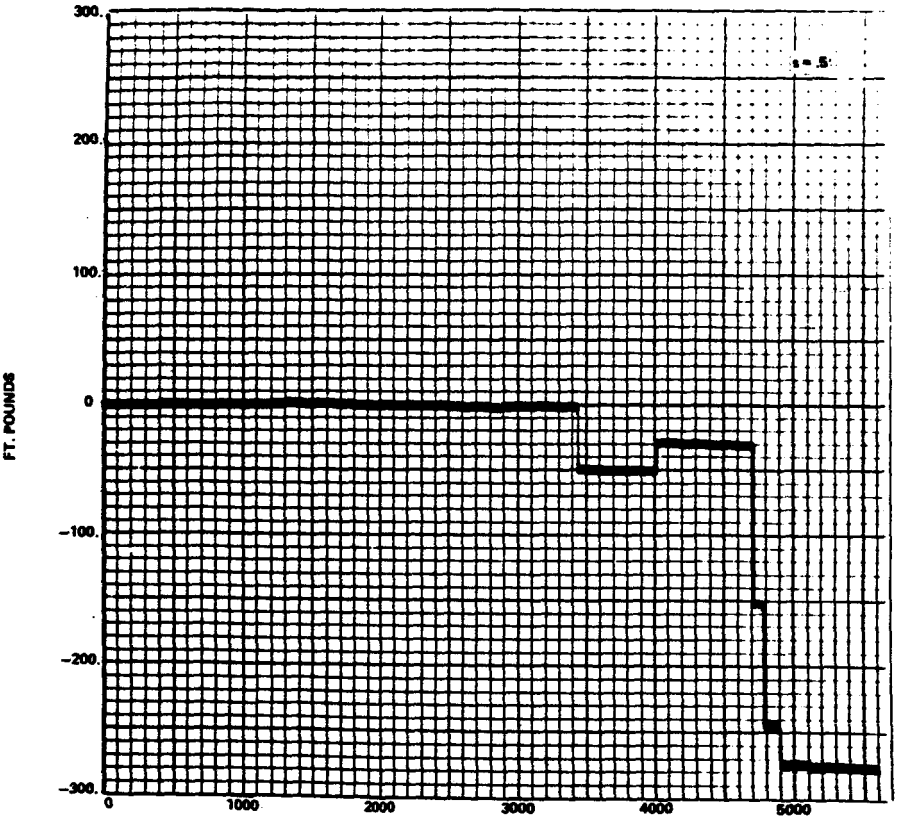
FIGURE 48 - GIMBAL ANGLES FOR $\Delta t = .2$

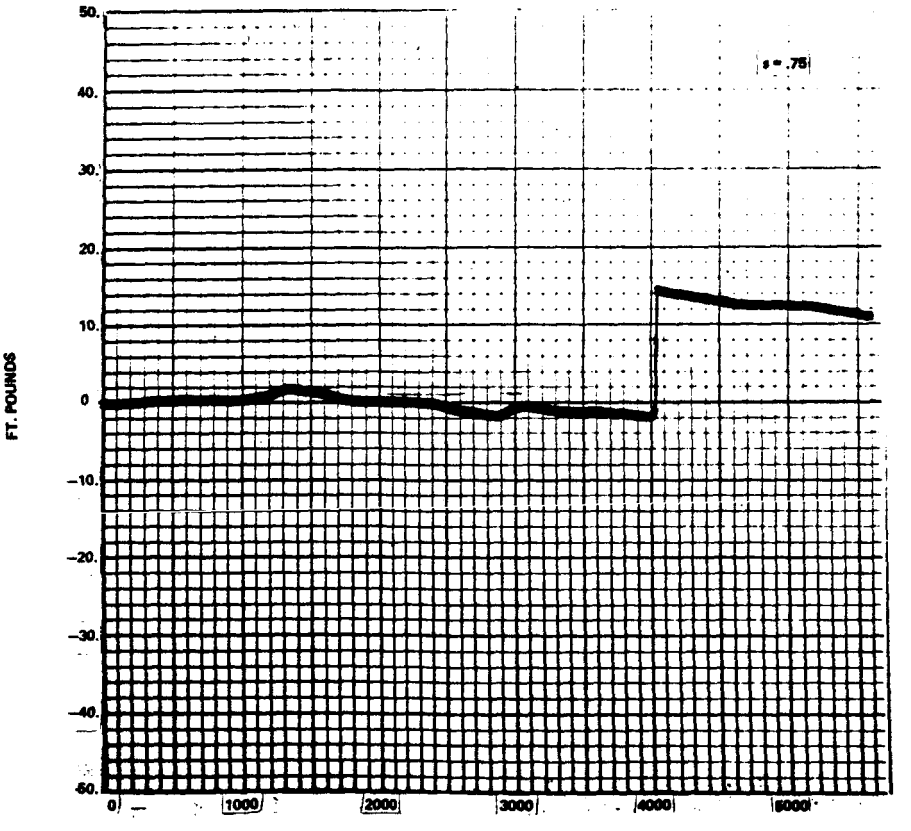




GYRO #3

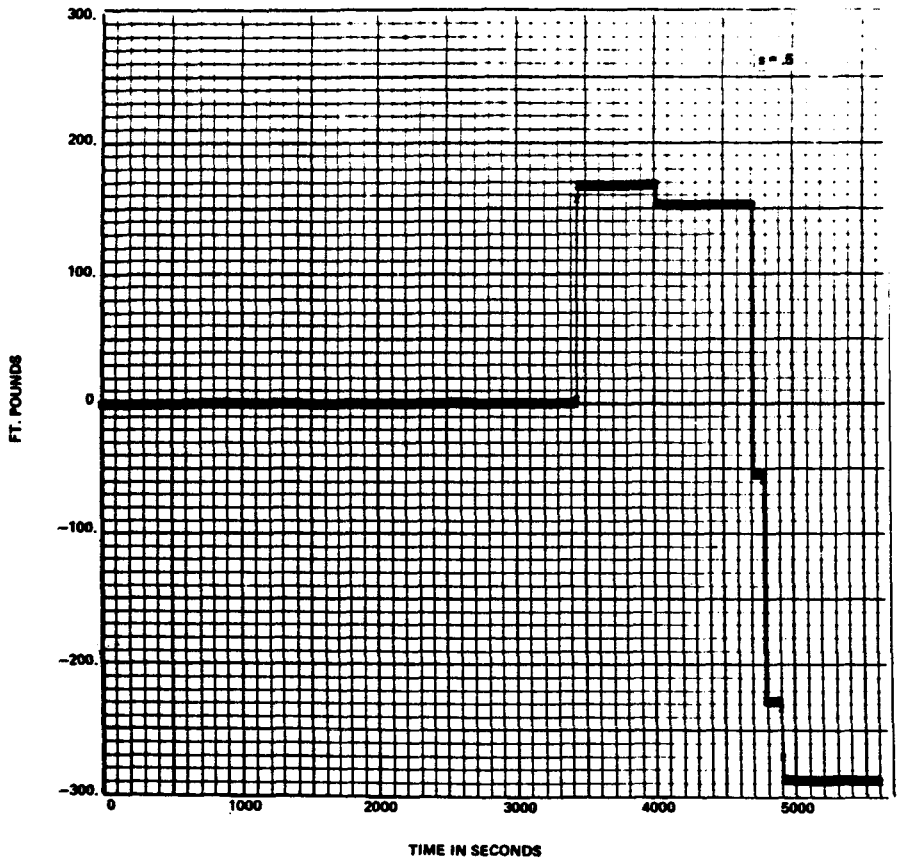
FIGURE 4C - GIMBAL ANGLES FOR $\Delta t = .2$

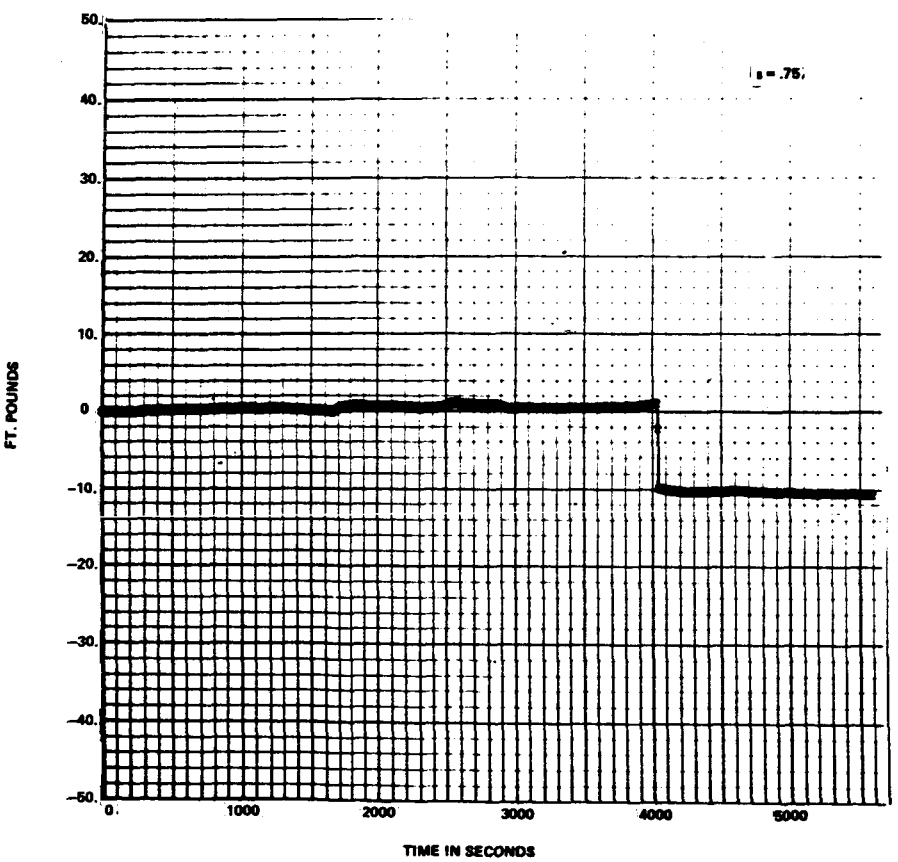




GYRO #1

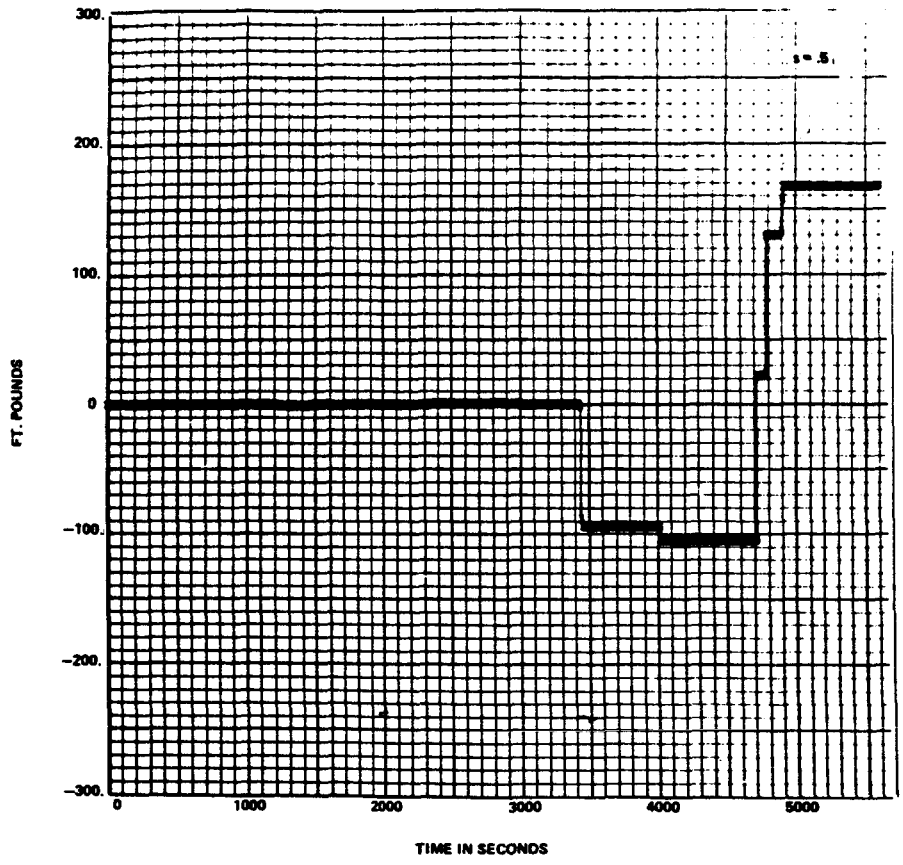
FIGURE 6A - TOTAL TORQUE ERRORS – $\Delta t = .2$

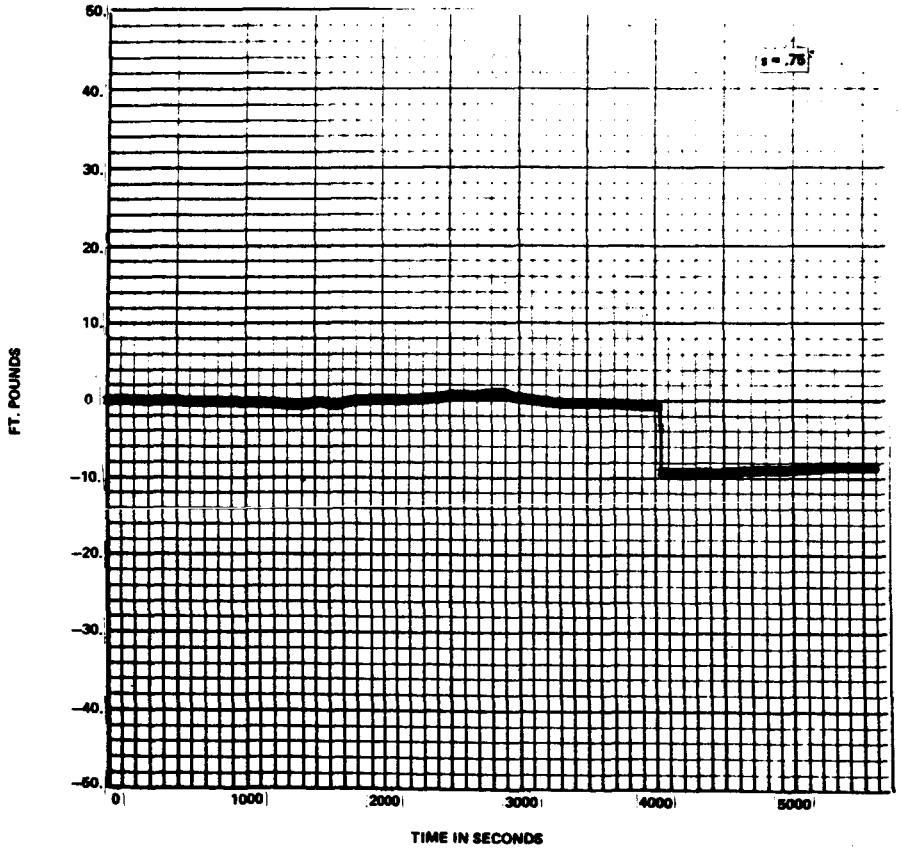




GYRO #2

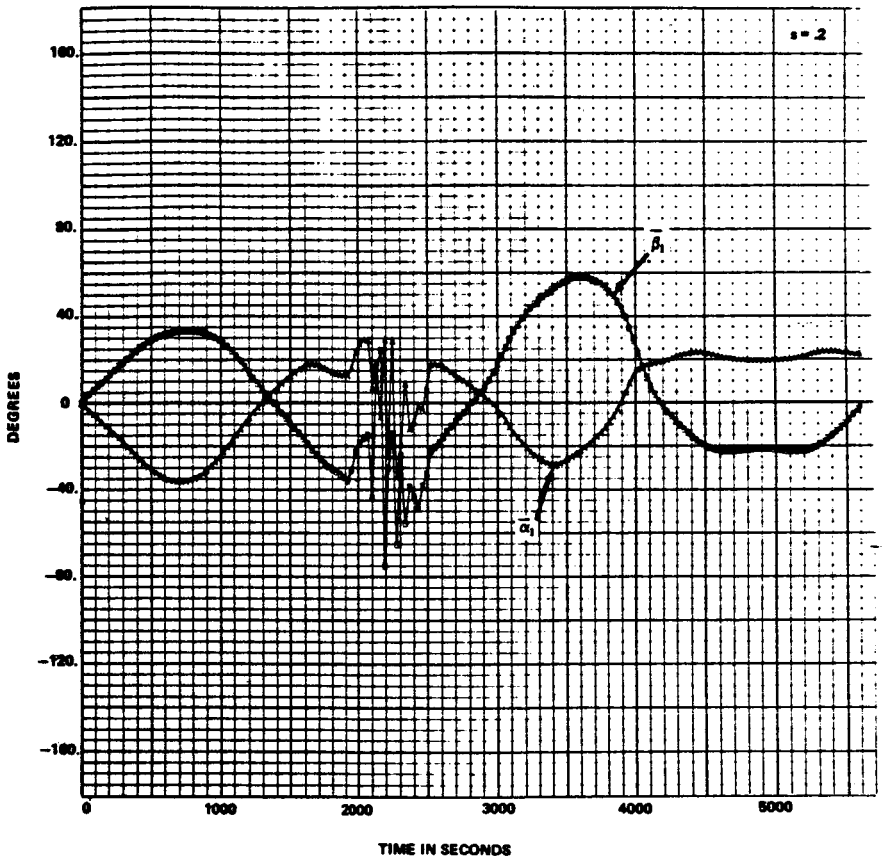
FIGURE 5B - TOTAL TORQUE ERRORS – $\Delta t = .2$

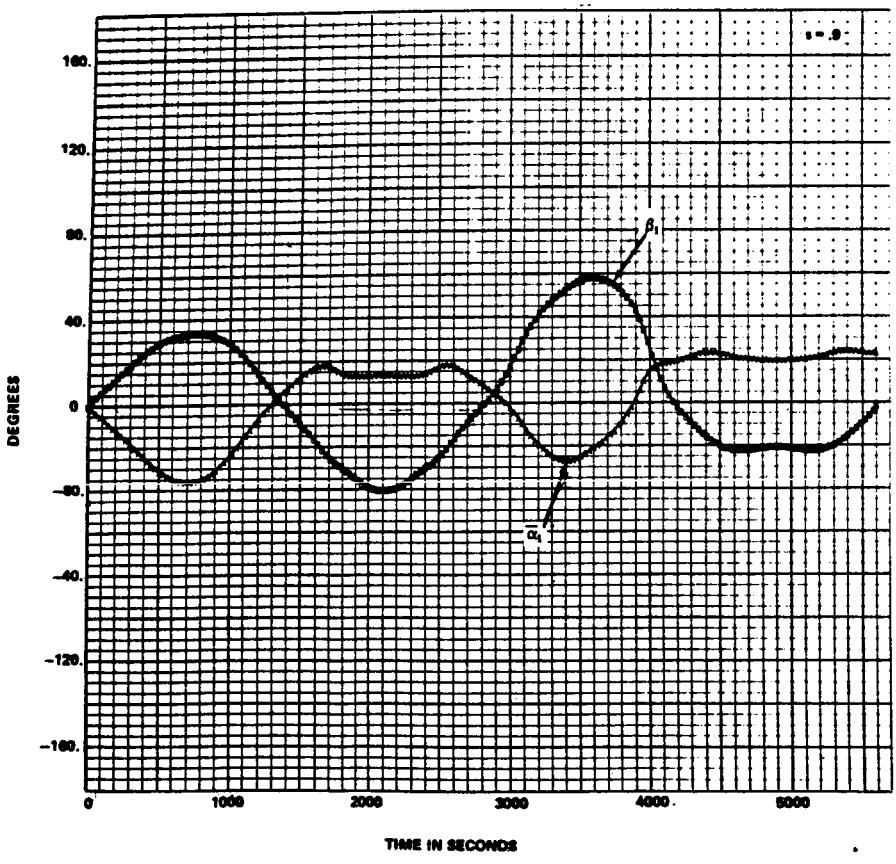




GYRO #3

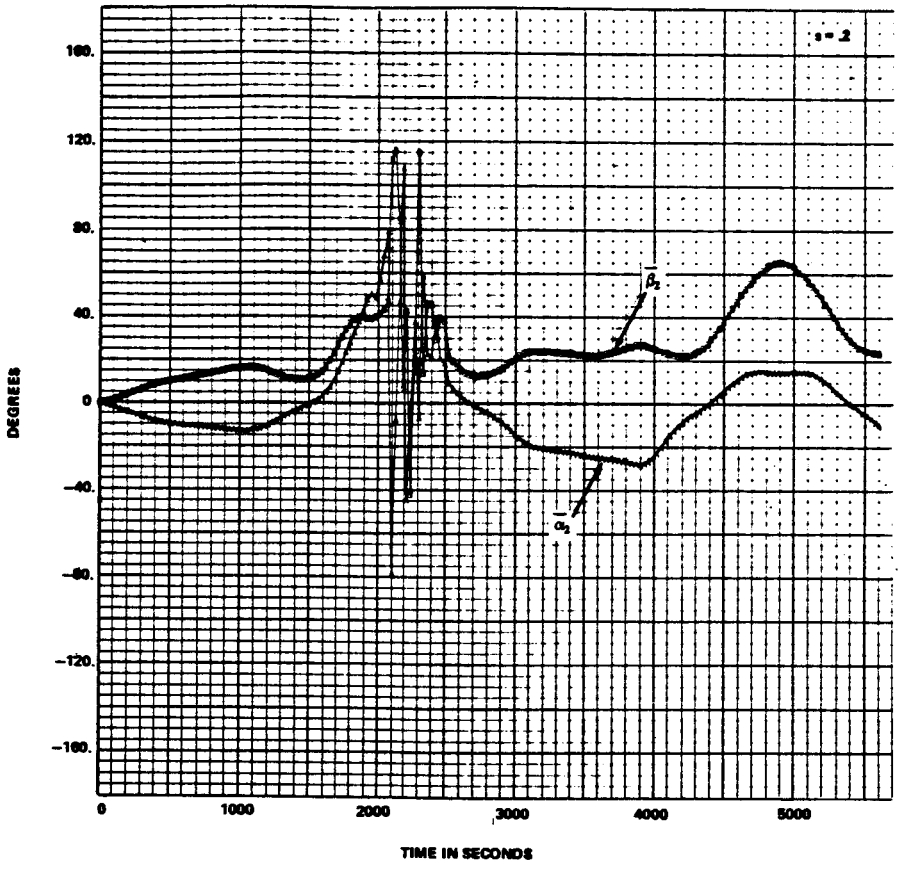
FIGURE 8C - TOTAL TORQUE ERRORS – $\Delta t = .2$

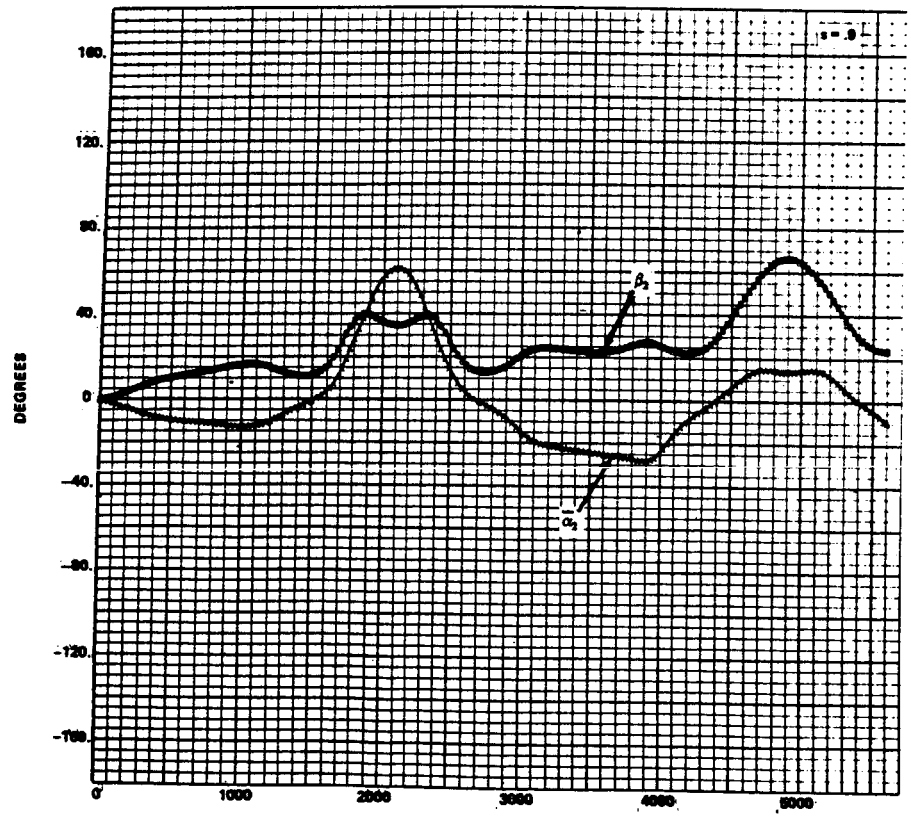




GYRO #1

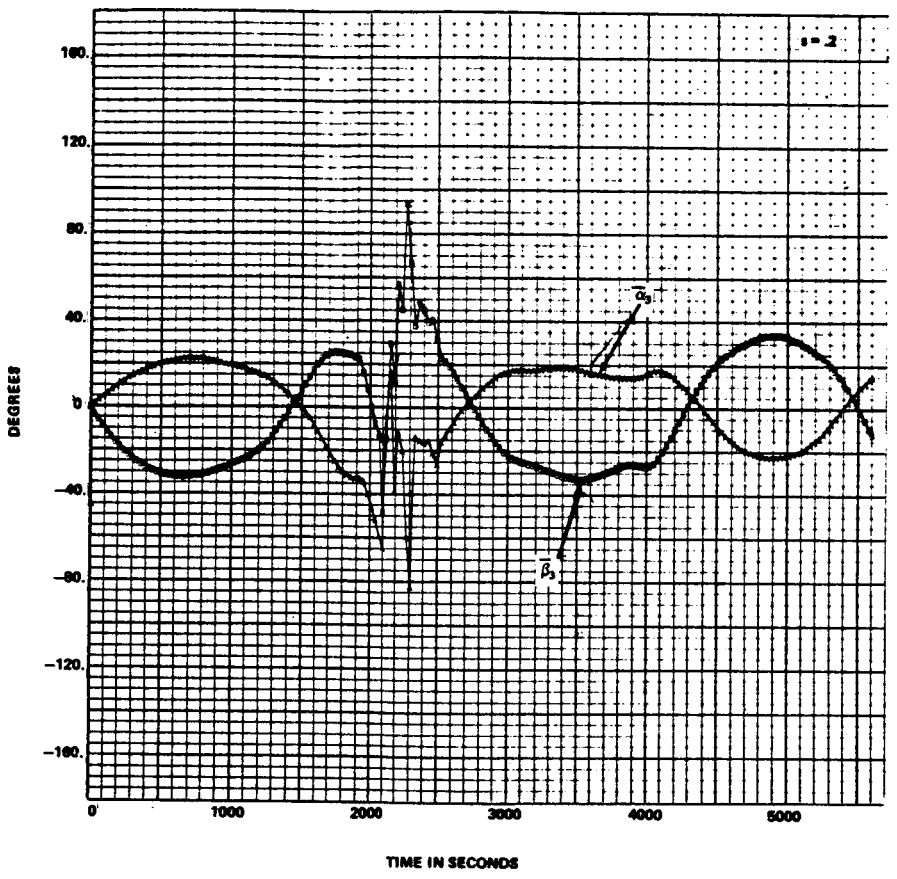
FIGURE 6A - GIMBAL ANGLES – $\Delta t = .2$

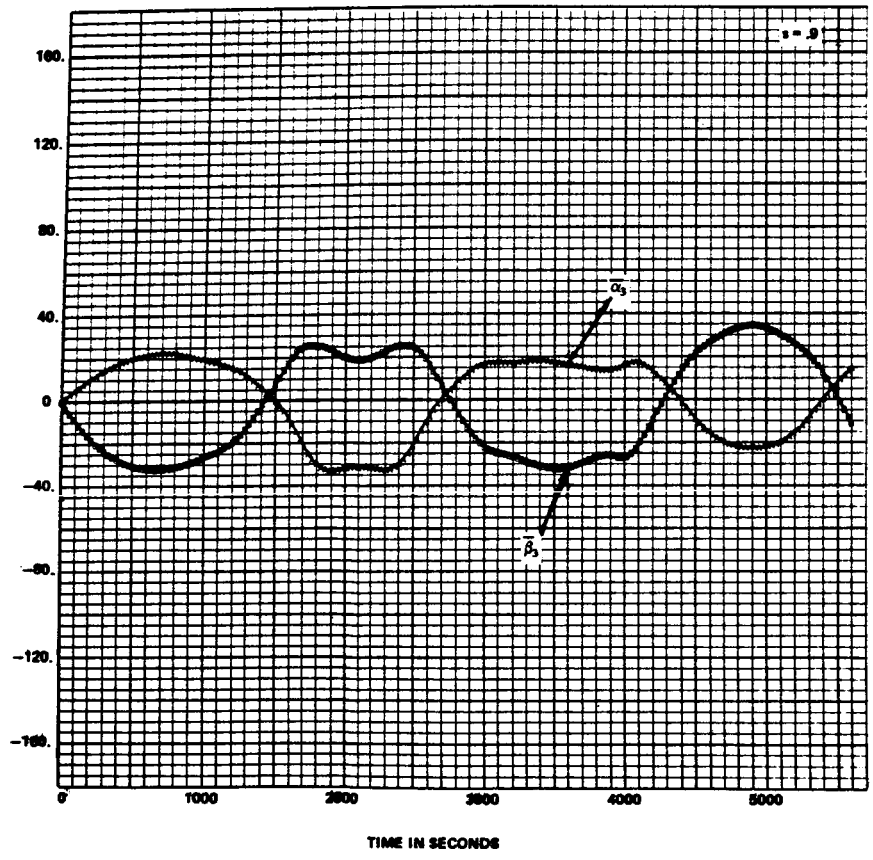




GYRO #2

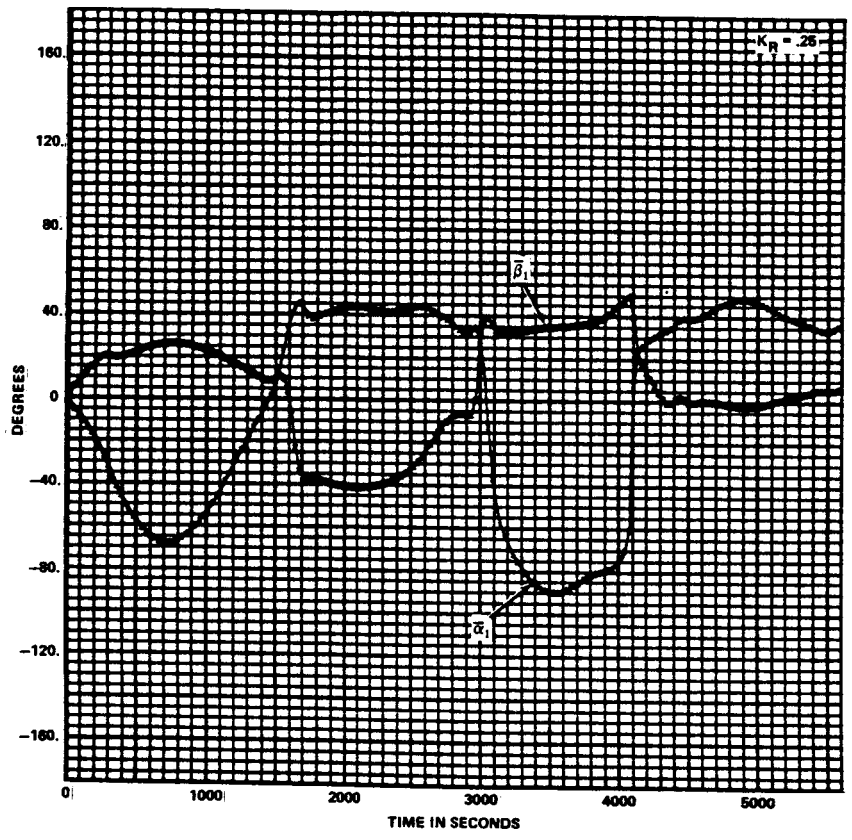
FIGURE 6B - GIMBAL ANGLES – $\Delta t = .2$

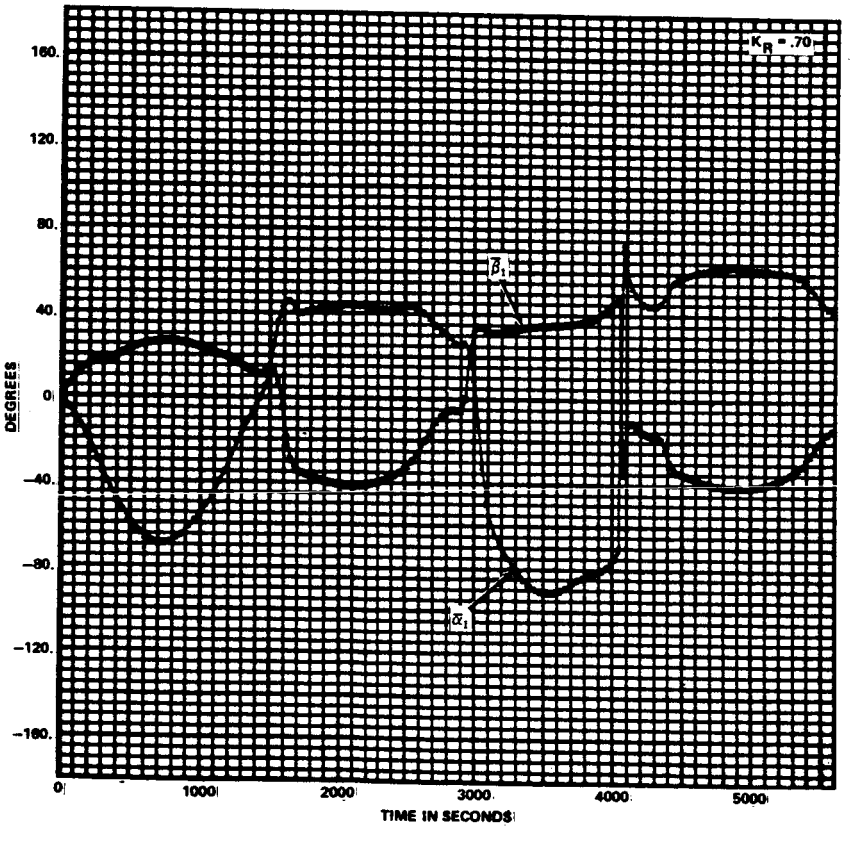




GYRO #3

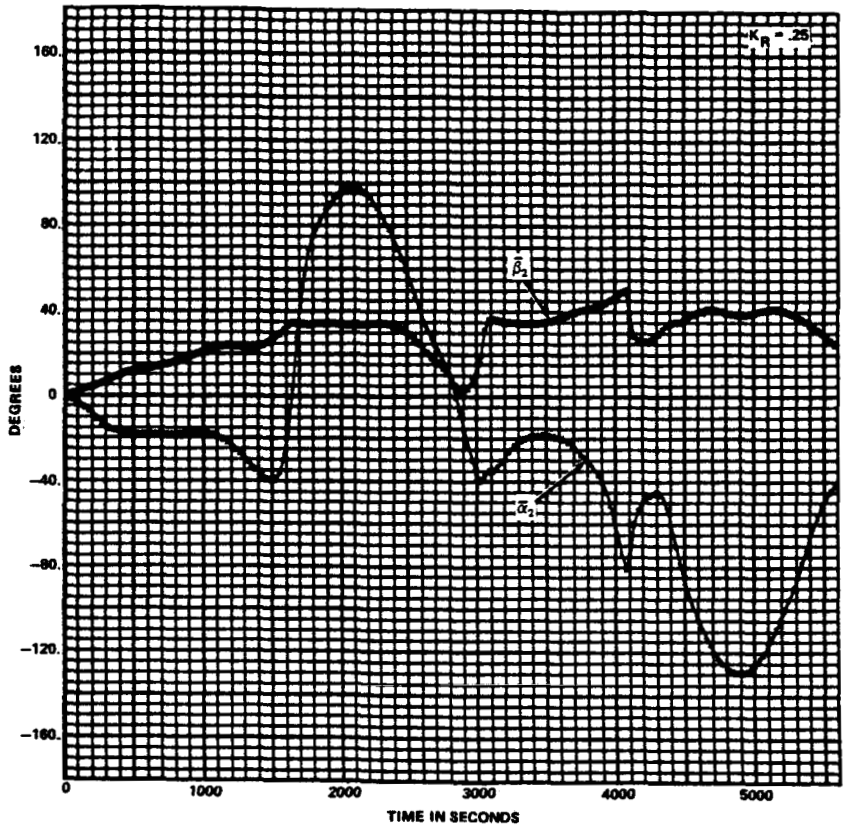
FIGURE 6C - GIMBAL ANGLES – $\Delta t = .2$

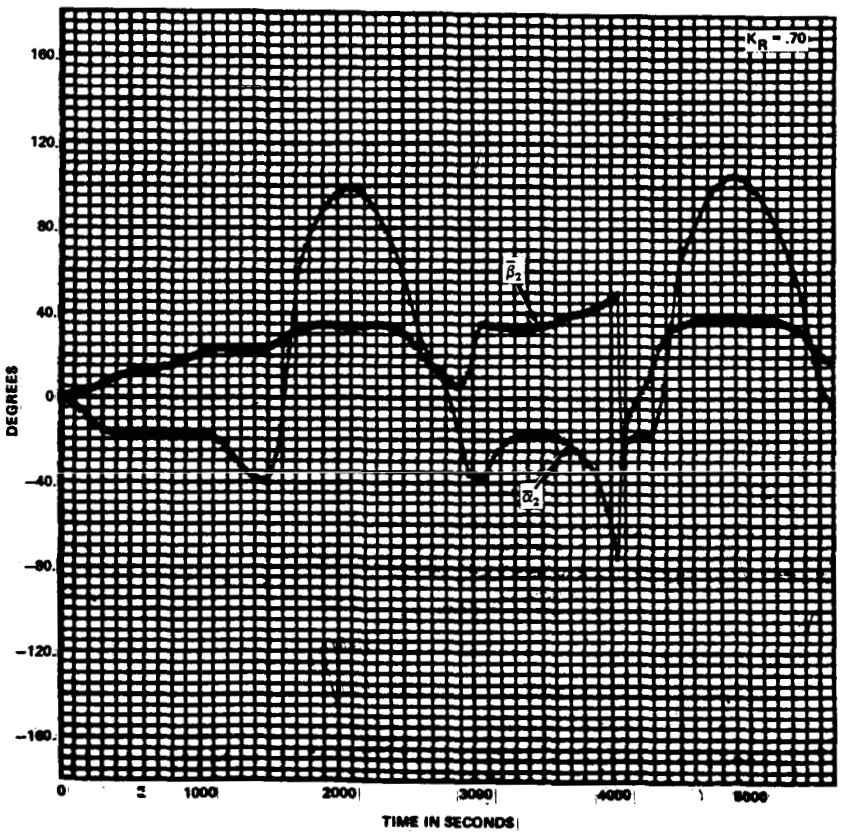




GYRO #1

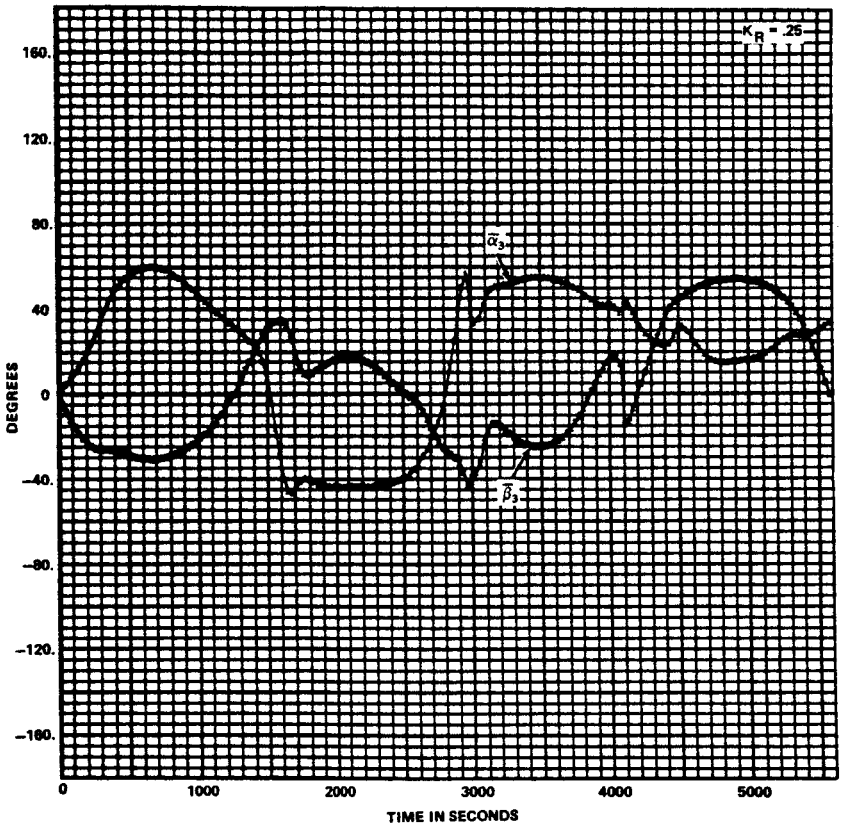
FIGURE 7A - GIMBAL ANGLES, MSFC LAW

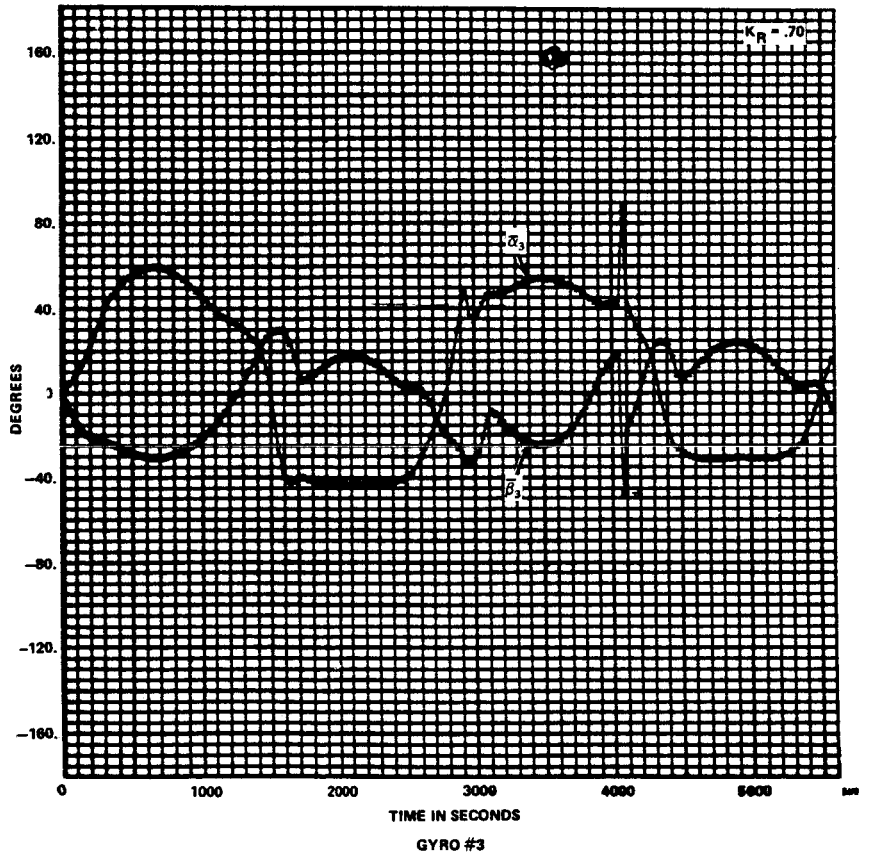




GYRO #2

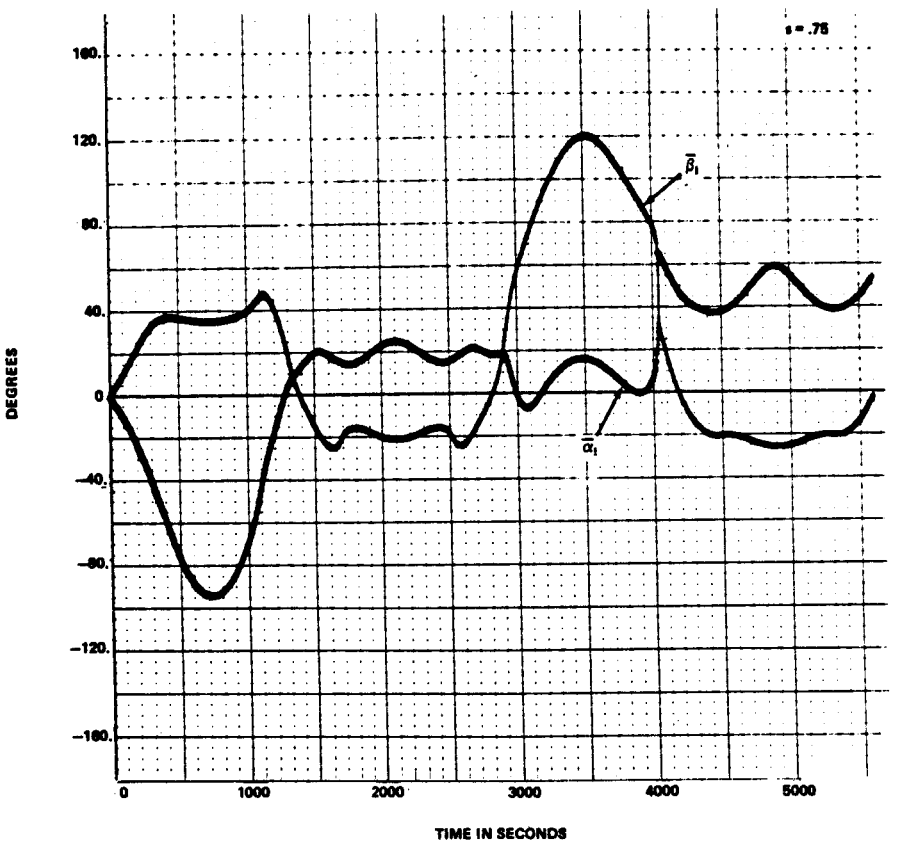
FIGURE 7B - GIMBAL ANGLES, MSFC, LAW

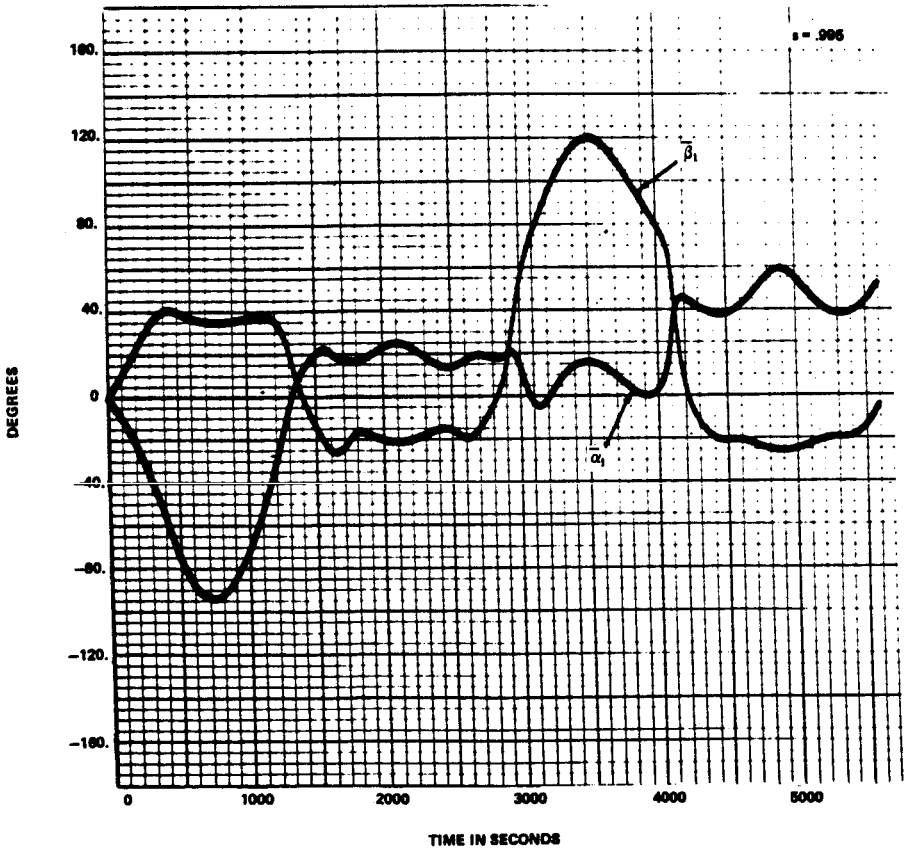




GYRO #3

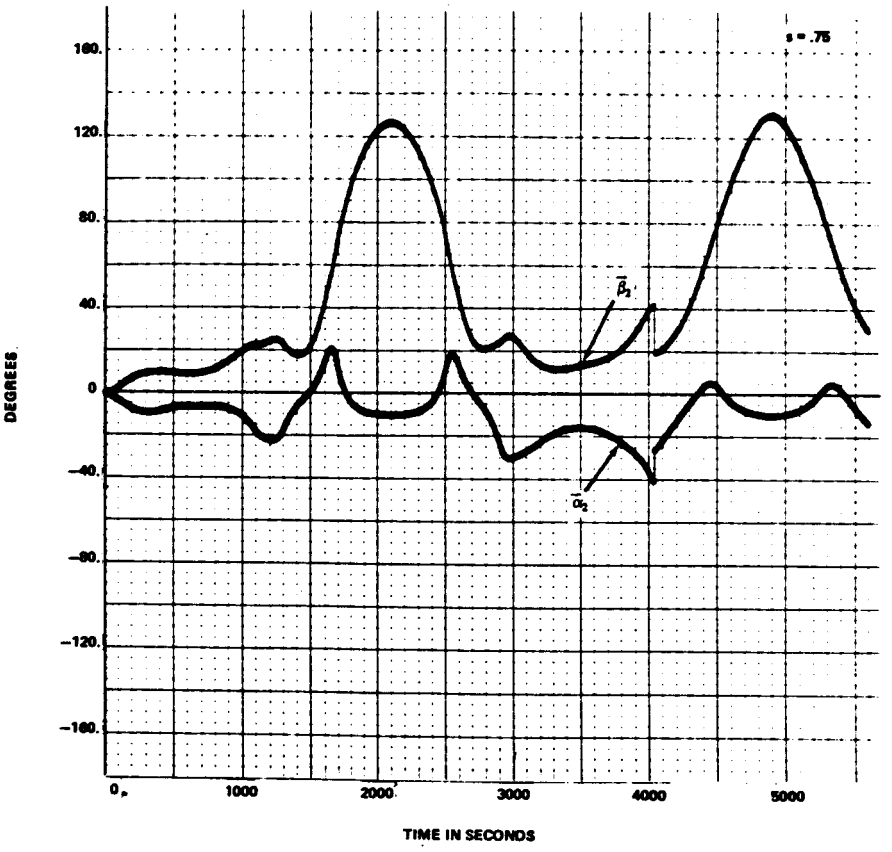
FIGURE 7C - GIMBAL ANGLES, MSFC, LAW

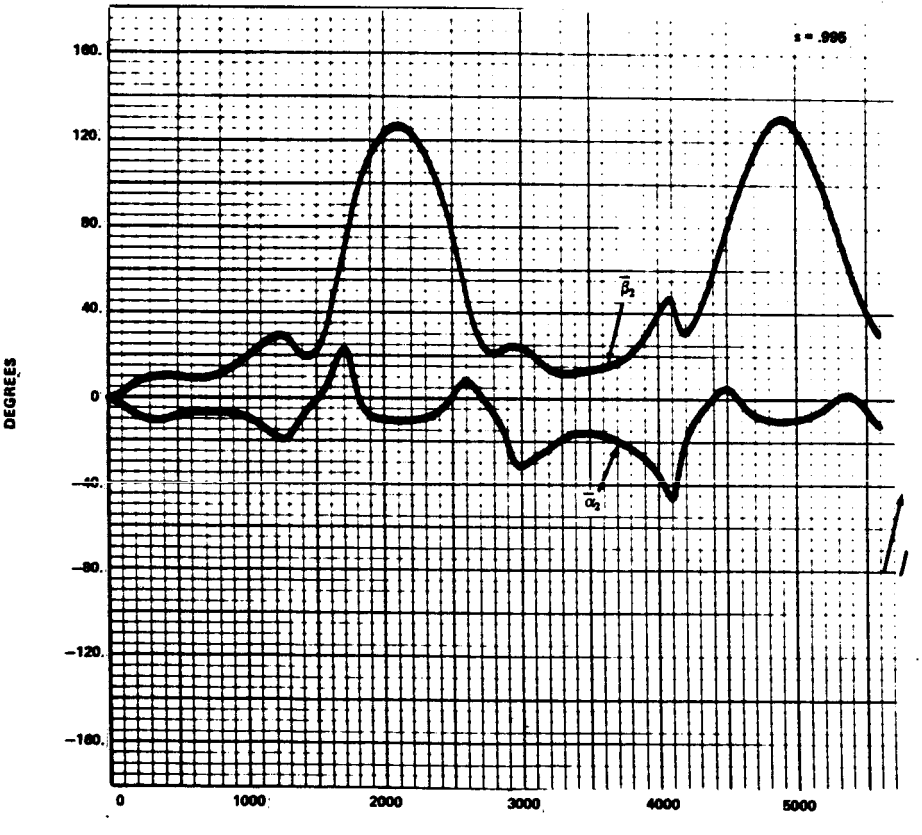




GYRO #1

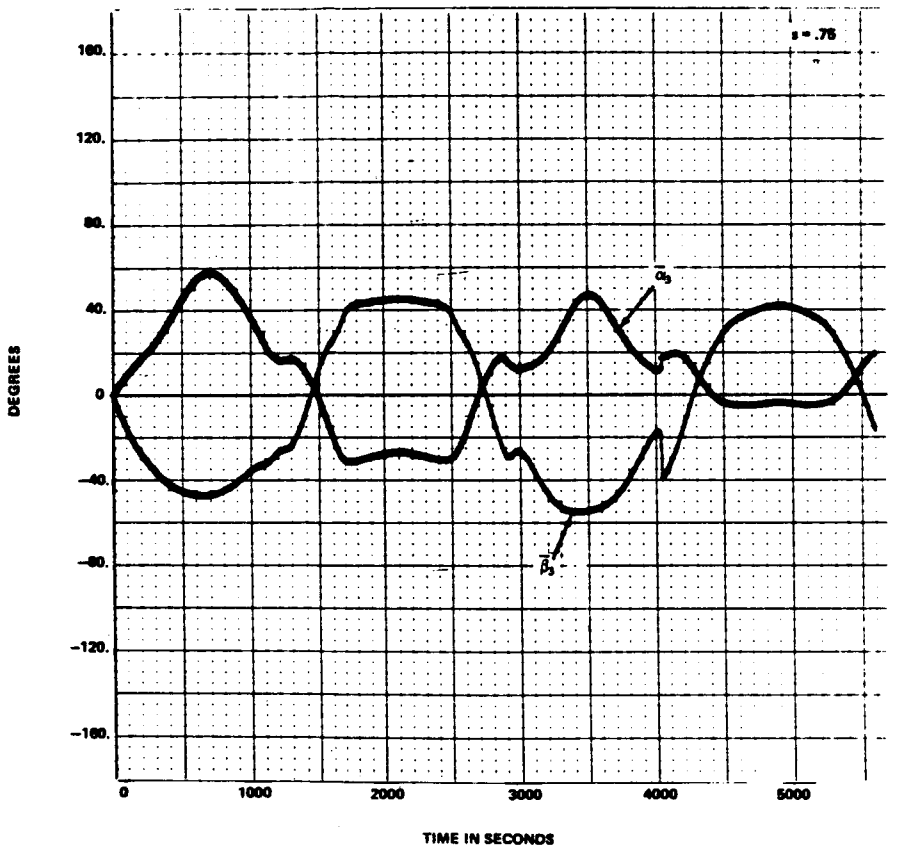
FIGURE 8A - GIMBAL ANGLES – $\Delta t = .2$

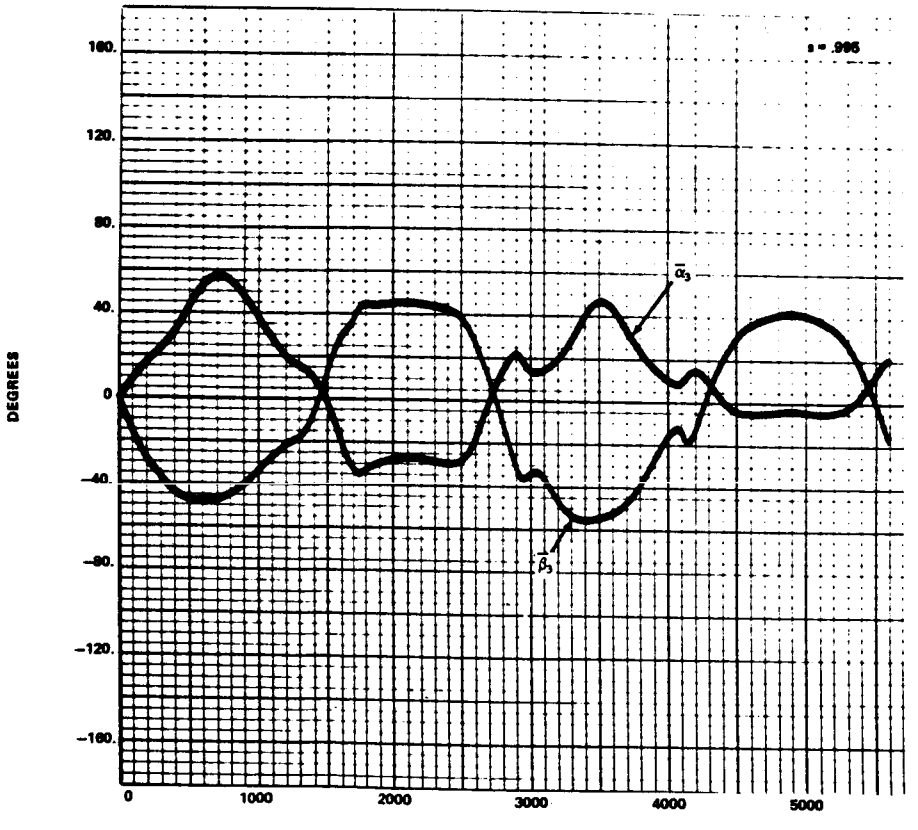




GYRO #2

FIGURE 88 - GIMBAL ANGLES – $\Delta t = .2$





GYRO #3

FIGURE 8C - GIMBAL ANGLES – $\Delta t = .2$